Computationally Efficient Multi-Asset Stochastic Volatility Modeling

by

Yizhou Fang

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Statistics

Waterloo, Ontario, Canada, 2018

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner:	Geneviève Gauthier Professor, Dept. of Decision Sciences, HEC Montréal
Supervisor(s):	Martin Lysy Associate Professor, Dept. of Statistics and Actuarial Science Don McLeish Professor, Dept. of Statistics and Actuarial Science
Internal Member:	Paul Marriott Professor, Dept. of Statistics and Actuarial Science Tony Wirjanto Professor, Dept. of Statistics and Actuarial Science
Internal-External Member:	Andrew Heunis Professor, Dept. of Electrical and Computer Engineering

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

I am the sole author of Chapters 1, 4 and 5.

Chapters 2 and 3 of this thesis have been prepared for submission to peer-reviewed journals. They have been edited by my supervisors Dr. Martin Lysy and Dr. Don McLeish. I contribute some ideas, all of the experiment designs and all of the data analysis. I also draft the initial manuscripts.

Abstract

Stochastic volatility (SV) models are popular in financial modeling, because they capture the inherent uncertainty of the asset volatility. Since assets are observed to co-move together, multi-asset SV (mSV) models are more appealing than combining single-asset SV models in portfolio analysis and risk management. However, the latent volatility process renders the observed data likelihood intractable. Therefore, parameter inference typically requires computationally intensive methods to integrate the latent volatilities out, so that it is computationally challenging to estimate the model parameters.

This three-part thesis is concerned with mSV modeling that is both conceptually and computationally scalable to large financial portfolios. In Part I, we explore the potential of substituting the latent volatility by an observable market proxy. For more than 20 years of out-of-sample predictions, we find that modeling the Standard and Poor's 500 (SPX) index by a simple framework of Seemingly Unrelated Regressions (SUR) with VIX volatility proxy is comparable to the benchmark Heston model with latent volatility, at a fraction of the computational cost.

In Part II, we propose a new mSV model structured around a common volatility factor, which also can be proxied by an observable process. Unlike existing mSV models, the number of parameters in ours scales linearly instead of quadratically in the number of assets – a desirable property for parameter inference of high-dimensional portfolios. Empirical evidence suggests that the common-factor volatility structure has considerable benefits for option pricing compared to a richer class of unconstrained models.

In Part III, we propose an approximate method of parameter inference for mSV models based on the Kalman filter. A large-scale simulation study indicates that the approximation is orders of magnitude faster than exact inference methods, while retaining comparable accuracy.

Acknowledgements

First, and most importantly, I want to express my sincere gratitude and appreciation to my supervisors Dr. Martin Lysy and Dr. Don McLeish. Without their patient guidances, this will not be possible.

I also want to thank my Examining Committee members, Dr. Tony Wirjanto, Dr. Paul Marriott, Dr. Andrew Heunis and Dr. Geneviève Gauthier. Thank you for your insightful suggestions.

The Department of Statistics and Actuarial Science is a wonderful family. I will always remember the delightful talks and coffee walks with the fellow Ph.D. candidates, especially my buddies Yoshi, Fei and Chunlin. The professors in this Department are all admirable. Thank you for your teaching. You set up examples of integrity and kindness. The administrative members make everything so easy, especially Mary Lou, Greg, Lisa, Helen and Carlos. Without you, my life of these years must be much harder.

Finally, I thank my parents and my grandmother for the unconditional support.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The Black-Scholes model (Black and Scholes, 1973) is the foundation of modern financial modeling. Let S_t denote the price of a given asset at time t. Then the dynamic of S_t is driven by a Geometric Brownian Motion, which is

$$\mathrm{d}S_t = \alpha S_t \mathrm{d}t + \sigma S_t \mathrm{d}B_t, \tag{1.1}$$

where α is the compound interest rate, σ is the so-called volatility of the asset on the standard deviation scale, and $\{B_t\}_{t\geq 0}$ is a Brownian motion.

We focus on two main features of the Black-Scholes model (1.1). First, it is a continuous time model based on a stochastic differential equation (SDE), so that this model can be applied conveniently to option pricing. Second, the asset volatility (σ) is set to be constant. While advantageous for analytic tractability, the assumption of constant volatility is not realistic. Figure 1.1 shows the daily log-return $r_t = \log(S_{t+1}) - \log(S_t)$, where $S_t = S(t \times \Delta\mathrm{T})$, $t = 0, \ldots, n$, and $\Delta\mathrm{T} = \frac{1}{252}$ years indicating daily observations, and the absolute log-return $|r_t|$ of the S&P500 (SPX) from 1990 to 2016. Under the Black-Scholes model (1.1), the daily log-returns have constant variance,

$$r_t \overset{\text{iid}}{\sim} \mathcal{N}\left((\alpha - \tfrac{1}{2}\sigma^2)\Delta\mathrm{T}, \sigma^2\Delta\mathrm{T}\right),$$

where $\Delta\mathrm{T} = \frac{1}{252}$ indicates daily observation. We can easily see that the variance of the SPX daily log-return process are time-varying and the peak log-returns cluster, especially

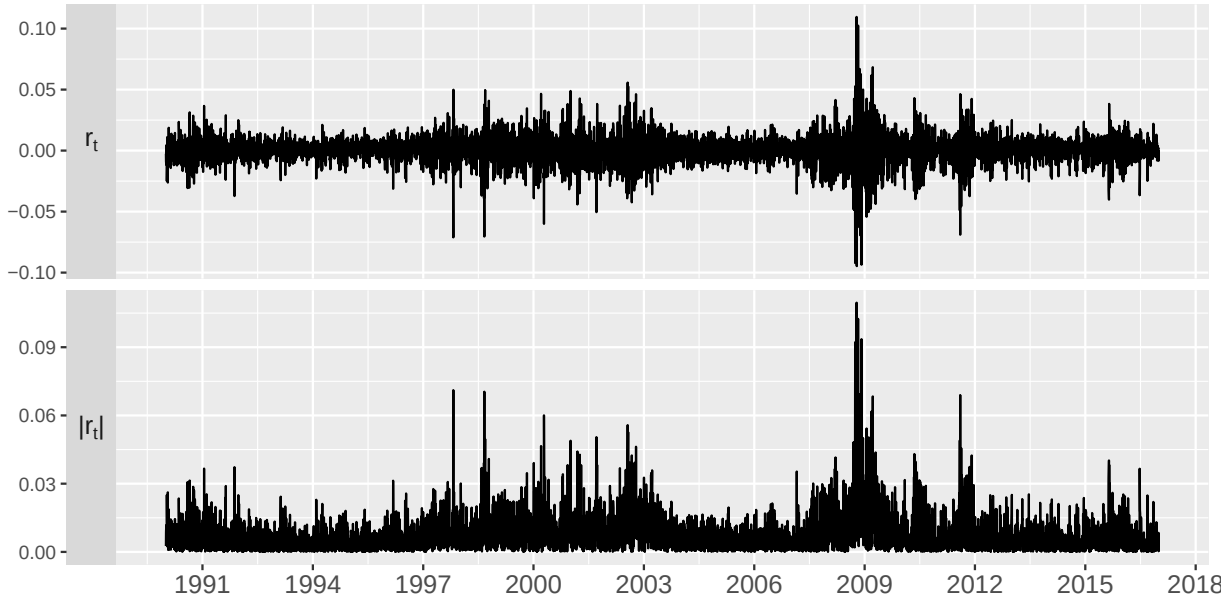


Figure 1.1: *Top:* SPX returns; *bottom:* SPX absolute returns.

during 2008. The presence of these features provides strong evidence that the volatility of the SPX is highly time-dependent, thus violating the constant variance assumption of the Black-Scholes model. Note that the SPX is a weighted average of the prices of various individual stocks, which means it should have less volatility comparing to the individual assets due to diversification of risks. Hence the time-dependence property of the volatility is likely to be inherited by individual assets as well.

Stochastic Volatility (SV) models are proposed to deal with the time-varying volatility process of the returns. Bollerslev et al. (1992) summarize a large body of literature on the empirical evidence of the stochastic nature of the asset return volatility and the correlation between the volatility and its underlying asset return. The pioneering works of Hull and White (1987); Scott (1987); Wiggins (1987) propose the SDE based single-asset SV models and focus on the advantages of the SV models over the Black-Scholes model on option pricing. Typically, a single-asset SV model is formed by a Geometric Brownian Motion with a time-varying volatility and an addition SDE, which models the volatility. The latter SDE incorporates randomness into the volatility process, hence the use of the term

stochastic. Among single-asset SV models, the Heston model (Heston, 1993) is perhaps the most famous, since it has a semi-analytical formula for European call options and its volatility process has some desirable features such as mean-reverting. The Heston model is driven by two SDEs:

$$\begin{aligned} \mathrm{d}S_t &= \alpha S_t \mathrm{d}t + S_t V_t \mathrm{d}B_t^S, \\ \mathrm{d}V_t^2 &= \gamma(\mu - V_t^2)\mathrm{d}t + \sigma V_t \mathrm{d}B_t^V, \end{aligned} \qquad \mathrm{cor}(B_t^S, B_t^V) = \rho, \tag{1.2}$$

where V_t is the volatility process on the standard deviation scale. As we can see, the asset process of the Heston model (1.2) is exactly the same as the Black-Scholes model, except that the volatility depends on time. The variance (V_t^2) process follows a Cox-Ingersoll-Ross (CIR) process (Cox et al., 1985), where $\gamma > 0$, $\sigma > 0$ and $2\gamma\mu > \sigma^2$. It guarantees that the variance reverts to the mean μ with the positive speed-of-mean reversion parameter γ. The larger γ indicates the mean reverting rate is faster and the autocorrelation of the volatility is smaller. The parameter σ called the "volatility-of-volatility". When the value of σ goes to 0, the dynamics are dominated by the drift term. As long as the parameters satisfy the condition of $2\gamma\mu > \sigma^2$, the upward drift dominates the process when the process approaches 0, so that it guarantees that the volatility process stays positive. The coefficient ρ captures the contemporary correlation between the movements of the asset price/return and its volatility. This coefficient being negative has been interpreted in empirical stochastic volatility literatures, e.g., Hull and White (1987); Harvey and Shephard (1996); Yu (2005); Ait-Sahalia et al. (2013), as capturing the "leverage effect", which is a financial phenomenon by which the decline in the asset price is usually associated with a high volatility in the contemporary or lagged period.

1.1 Literature Review

Estimation of SV models to financial data consists of estimating the SV parameters θ based on discrete observations of the asset price $\boldsymbol{S} = (S_0, \ldots, S_n)$, where $S_t = S(t \times \Delta\mathrm{T})$, $t = 0, \ldots, n$. We define $\boldsymbol{V} = (V_0, \ldots, V_n)$ similarly. For most SV models this is a formidable

computational task. To see this, consider the likelihood function on the observed data:

$$\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{S}) = \int p(\boldsymbol{S}, \boldsymbol{V}|\boldsymbol{\theta}) d\boldsymbol{V} = \int \prod_{t=1}^{n} p(S_t, V_t|S_{t-1}, V_{t-1}, \boldsymbol{\theta}) d\boldsymbol{V}, \tag{1.3}$$

where the full likelihood $p(\boldsymbol{S}, \boldsymbol{V}|\boldsymbol{\theta})$ can be obtained, but the latent part $\boldsymbol{V}$ must be integrated out. Thus, inference for Heston's and other SV models involves two challenges: (i) the transition density of the SDE model $p(S_t, V_t|S_{t-1}, V_{t-1}, \boldsymbol{\theta})$ is not available in closed-form and (ii) the volatilities are unobserved and therefore must be integrated out.

The first issue is typically addressed by that of replacing the transition density in (1.3) by the Euler-Maruyama approximation (Maruyama, 1955). For the Heston model (1.2), the Euler-Maruyama apparoximation is

$$\begin{aligned} S_{t+1} - S_t &= \alpha S_t \Delta\mathrm{T} + S_t V_t \epsilon_t^S, \\ V_{t+1}^2 - V_t^2 &= \gamma(\mu - V_t^2)\Delta\mathrm{T} + \sigma V_t \epsilon_t^V, \end{aligned} \qquad \begin{pmatrix} \epsilon_t^S \\ \epsilon_t^V \end{pmatrix} \overset{\text{iid}}{\sim} \mathcal{N}\left(\mathbf{0}, \Delta\mathrm{T} \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right),$$

which strongly converges to the exact transition density on $\sqrt{\Delta\mathrm{T}}$. An even better approximation can be yielded by the Milstein scheme (Mil'shtejn, 1975), which converges strongly on $\Delta\mathrm{T}$. In practice, the Euler-Maruyama approximation is often used and empirical evidence indicates that this approximation is generally usable for financial data recorded at daily intervals (Phillips and Yu, 2009). If we use Itō's lemma to change the model variables so that they have constant diffusion functions, the Euler-Maruyama method is actually accurate as the Milstein scheme (Mil'shtein, 1979; Pedersen, 1995)

In addition, there exist SV models defined purely in discrete-time, e.g., that of Harvey and Shephard (1996):

$$\begin{aligned} r_t &= \beta + e^{h_t/2}\epsilon_t, \\ h_{t+1} &= \alpha_0 + \alpha_1 h_t + \sigma\eta_t, \end{aligned} \qquad \begin{pmatrix} \epsilon_t \\ \eta_t \end{pmatrix} \overset{\text{iid}}{\sim} \mathcal{N}_2\left(\mathbf{0}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right), \tag{1.4}$$

where r_t is the log-variance and h_t is the log-volatility. This model allows non-zero correlation ρ, so that it is a generalization of the model proposed by Harvey et al. (1994), which has zero correlation ρ.

While the model likelihood $\mathcal{L}(\boldsymbol{\theta}|r_0, \ldots, r_n, h_0, \ldots, h_n)$ for the discrete time SV model (1.4) can be written in closed-form, the log-volatility is still latent and has to be integrated out

of the model likelihood function. A further simplification which does not use latent data is the class of generalized autoregressive conditional heteroscedasticity (GARCH) models. The work of Engle (1982) and Bollerslev (1986) lay the foundation of the GARCH-type models, which have been very popular in financial time-series modeling ever since. A simple GARCH(1,1) model can be expressed as

$$\begin{aligned} r_t &= \mu + \sigma_t z_t, \\ \sigma_t^2 &= \alpha_0 + \alpha_1 r_{t-1}^2 + \beta_1 \sigma_{t-1}^2, \end{aligned} \qquad z_t \overset{\text{iid}}{\sim} F(z),$$

where μ is the drift rate, r_t is the log-return, σ_t is the volatility, and the only randomness is the white noise process z_t (which for identifiability purposes has mean 0 and variance 1). The distribution of z_t is often set to, but does not need to be, a Gaussian. A t-distribution (Bollerslev, 1987) or a normal-inverse Gaussian distribution (Barndorff-Nielsen, 1997) can also be used to accommodate the heavy-tailed and/or skewed return distributions. Instead of assuming that there is an uncertain component in the volatility process, a GARCH-type model treats the volatility process as predictable since the current volatility is a deterministic function of previous log-returns and previous volatilities. Hence, GARCH-type models are not considered to be SV models in the context of this dissertation.

Often, discrete-time SV models, for example the model (1.4), can be viewed as a first order approximation of a stochastic Taylor expansion to an SDE. While the same is true for GARCH models, there might exist multiple continuous time limits: different limits lead to different SDEs, driven by either one or two Brownian motions depending on the particular limiting approach (Nelson, 1990; Drost and Werker, 1996; Corradi, 2000). In whichever case, the GARCH corresponds to a Heston-like model with parameter restrictions, notably $\rho = 0$ in (1.2). Thus, they have less flexibility to fit return data and consequently produce less accurate forecasts than SV models with latent volatility (Kim et al., 1998; Fleming and Kirby, 2003). In addition, the option pricing of the model depends on the non-zero ρ (Hull and White, 1987; Harvey and Shephard, 1996; Yu, 2005). Duan (1995) discuss the local risk neutral measure of one specific GARCH model but it lacks an exact closed-form option pricing scheme. Heston and Nandi (2000) propose a GARCH model with a Heston model limit, which follows an analytical solution of option pricing according to the Heston

model limit. In other words, the SV models seem to be more convenient for option pricing than the GARCH-type models. In addition, the volatility in a GARCH-type model is predictable by previous information, however, modeling the volatility stochastically as the SV models is theoretically more appealing (Fleming and Kirby, 2003).

Until now, we have focused exclusively on single-asset models. However, multi-asset models are also crucial for financial portfolio management, since closely related assets such as stocks, exchange rates, and derivative prices are typically observed to fluctuate together (Aydemir, 2002). In order to construct, evaluate and hedge a portfolio, it is very important to learn about the joint distribution of asset returns. In addition, the accurate estimation of the covariance matrix of multiple asset returns help investors better understand the risks of a particular portfolio. Risk management and portfolio optimization should be conducted within a multi-asset modeling framework, rather than combining results of multiple single-asset analyses (Liesenfeld and Richard, 2003).

Historically, multi-asset GARCH models precede mSV models; for example, consider the Constant Correlation Coefficient model of (Bollerslev, 1990)

$$\begin{aligned}
\boldsymbol{r}_t &= \mathrm{E}(\boldsymbol{r}_t|\boldsymbol{r}_{0:t-1}) + \boldsymbol{\epsilon}_t, \\
\mathrm{Var}(\boldsymbol{\epsilon}_t) &= \boldsymbol{\Sigma}_t \boldsymbol{R} \boldsymbol{\Sigma}_t, \\
\boldsymbol{\Sigma}_t &= \mathrm{diag}(\sigma_{1,t}, \ldots, \sigma_{q,t}),
\end{aligned}$$

where $\boldsymbol{R}$ is a constant correlation matrix and $\sigma_{i,t}$ is the volatility of the ith asset at time t which follows a GARCH structure, e.g., GARCH(1,1):

$$\sigma_{i,t}^2 = \omega_i + \alpha_i \epsilon_{i,t-1}^2 + \beta_i \sigma_{i,t-1}^2, \quad i = 1, \ldots, q.$$

Based on this CCC model, (Engle, 2002) proposes the Dynamic Conditional Correlation (DCC) model, which has a time-vary correlation matrix.

One of the earliest multi-asset SV model is introduced by Harvey et al. (1994) and has the form

$$\begin{aligned}
r_{it} &= \exp(\tfrac{1}{2} h_{i,t}) \epsilon_{it}, \\
h_{i,t+1} &= \alpha_i + \beta_i h_{i,t} + \sigma_i \eta_{it},
\end{aligned} \qquad \begin{pmatrix} \boldsymbol{\epsilon}_t \\ \boldsymbol{\eta}_t \end{pmatrix} \sim \mathcal{N}_{2q} \left(\mathbf{0}, \begin{pmatrix} \boldsymbol{\Sigma}_{1,1} & \boldsymbol{\Sigma}_{1,2} \\ \boldsymbol{\Sigma}_{2,1} & \boldsymbol{\Sigma}_{2,2} \end{pmatrix} \right), \tag{1.5}$$

where $\boldsymbol{\eta}_t = (\eta_{1t}, \ldots, \eta_{qt})$, $\boldsymbol{\epsilon}_t = (\epsilon_{1t}, \ldots, \epsilon_{qt})$, and $\boldsymbol{r}_t = (r_{1t}, ..., r_{qt})$ is the return vector of q assets on day t. Notice that this mSV model (1.5) is a natural extension of the single-asset SV model (1.4), and Harvey et al. (1994) set $\boldsymbol{\Sigma}_{1,2}$, $\boldsymbol{\Sigma}_{2,1}$ to zero matrices for computational convenience. This model has deep roots in the CCC model as the model structures are very similar. The difference is whether there exists a random part in the volatility modeling. Similar models with non-zero $\boldsymbol{\Sigma}_{1,2}$ and $\boldsymbol{\Sigma}_{2,1}$ are investigated by e.g., Chan et al. (2006); Ishihara and Omori (2012). Jacquier et al. (1995, 1999) propose a mSV model which assumes the returns can be explained by several factors, so that not the asset returns but the factors of the returns have their individual volatilities. The time-varying correlation mSV model investigated by Yu and Meyer (2006) is an extension of the DCC model. Another approach to obtain the time-vary correlation models is by employing the Wishart matrix process to model the variance matrix of the returns (Uhlig, 1997; Philipov and Glickman, 2006).

1.2 Contributions of this Thesis

As we discussed in Section 1.1, to obtain the model likelihood by the integration (1.3) poses a computational challenge for the inference of the Heston model (1.2). This challenge is shared by all of the SV models that have latent volatilities. To evaluate the integration (1.3), Markov chain Monte Carlo (MCMC) methods (e.g., Eraker, 2001; Broto and Ruiz, 2004; Golightly and Wilkinson, 2006; Andrieu et al., 2010; Kou et al., 2012; Beskos et al., 2013; Bladt et al., 2016; Kastner et al., 2017) and particle filters (e.g., Liu and Chen, 1998; Doucet et al., 2000; Andrieu and Doucet, 2002; Bates, 2006; Hurn et al., 2015) are often used.

Additionally, mSV models can have a very large number of parameters and considerable care must be taken for covariance matrix estimators to be positive-definite given the model's identifiability constraints. For example, Chan et al. (2006) find algebraic constraints on a single element of the covariance matrix for all other elements (above the diagonal) held fixed. Asai and McAleer (2009) guarantee positive-definiteness by generating the covariance matrix from a Wishart distribution, and Lopes et al. (2012) parametrize the covariance via

its Cholesky factors. However, for all models above, the number of covariance parameters scales $\mathcal{O}(q^2)$, where q is the number of assets. This is an undesirable feature for estimating mSV models to a large number of assets.

In this thesis, we make several contributions towards multi-asset stochastic volatility modeling that are both conceptually and computationally scalable to high-dimensional financial portfolios. Central to our work is the idea that considerable information about the latent stochastic volatilities can be extracted from observable market proxies. A simple example of such a proxy would be the VIX, which is calculated by the Chicago Board Options Exchange (CBOE) from option prices on the SPX, providing a popular measure of the average implied volatility of SPX for the upcoming 30 calendar days. Recent discussions of the importance of VIX in financial modeling are e.g., Ahoniemi (2006); Jiang and Tian (2007); Blair et al. (2001).

In Chapter 2, we consider modeling the SPX with the VIX as the observed volatility proxy. Specifically, we consider a discrete-time SV model with Gaussian innovations, simply replacing the latent volatility with the VIX proxy. Recasting this model in the framework of Seemingly Unrelated Regressions (SUR) (Zellner, 1962), we adapt an efficient Bayesian algorithm for parameter inference and forecasting of asset prices. We introduce an empirical transformation to fit non-Gaussian data, while retaining both the speed and exactness of the SUR approach. Daily out-of-sample predictions for more than 20 years of SPX data indicate that forecasting results of the proposed model are similar to that of the Heston model but more accurate than the GARCH(1,1) model. However, direct replacement of the latent volatility by the VIX is less effective for individual stocks, as individual volatility dynamics often differ substantively from those of the market average.

In Chapter 3, we propose a new SDE-based mSV model. Rather than replacing the latent volatilities with VIX, the proxy is included via a common-factor model, suggested by considerable empirical evidence of volatility factors (Engle and Figlewski, 2015; Herskovic et al., 2016). The resulting correlation structure on the model innovations is both interpretable and parsimonious, with the number of parameters scaling linearly in the number of assets. Analyzing a $q = 4$ asset portfolio between 2013-2017, we find that the common-factor volatility approach has considerable benefits for option pricing, with little loss in forecasting accuracy compared to unconstrained-correlation models.

Throughout these chapters, exact inference for SDE-based mSV models is conducted by Bayesian data augmentation. For example with the Heston model, this consists of choosing a prior $\pi(\boldsymbol{\theta})$ for the model parameters and using MCMC to sample from the augmented distribution

$$p(\boldsymbol{V}, \boldsymbol{\theta}|\boldsymbol{S}) \propto \pi(\boldsymbol{\theta}) p(\boldsymbol{S}, \boldsymbol{V}|\boldsymbol{\theta}) = \pi(\boldsymbol{\theta}) \prod_{t=1}^{n} p(S_t, V_t|S_{t-1}, V_{t-1}, \boldsymbol{\theta}),$$

where the transition densities are approximated by the Euler-Maruyama method. The specific algorithm is provided by the R package *msde* (Lysy and Tong, 2017) and is detailed in Appendix B. However, more efficient inference methods can be achieved by filtering the latent volatilities through the model's Markov structure. That is, suppose that for fixed $\boldsymbol{\theta}$, we have some ways of storing the conditional volatility distribution $p_{\boldsymbol{\theta}}(V_{t-1}|\boldsymbol{S}_{1:t-1}) := p(V_{t-1}|S_1, \ldots, S_{t-1}, \boldsymbol{\theta})$ and we could use it to calculate both

$$p_{\boldsymbol{\theta}}(S_t|\boldsymbol{S}_{1:t-1}) = \int p_{\boldsymbol{\theta}}(S_t|S_{t-1}, V_{t-1}) p_{\boldsymbol{\theta}}(V_{t-1}|\boldsymbol{S}_{1:t-1}) dV_{t-1}$$

and

$$\begin{aligned} p_{\boldsymbol{\theta}}(V_t|\boldsymbol{S}_{1:t}) &= \int p_{\boldsymbol{\theta}}(V_t, V_{t-1}|\boldsymbol{S}_{1:t}) dV_{t-1} \\ &= \frac{\int p_{\boldsymbol{\theta}}(S_t, V_t|S_{t-1}, V_{t-1}) p_{\boldsymbol{\theta}}(V_{t-1}|\boldsymbol{S}_{1:t-1}) dV_{t-1}}{p_{\boldsymbol{\theta}}(S_t|\boldsymbol{S}_{1:t-1})}. \end{aligned}$$

The latter provides the updated filtering density $p_{\boldsymbol{\theta}}(V_t|\boldsymbol{S}_{1:t})$ so that this algorithm can be iterative on $t = 1, \ldots, n$. Thus the likelihood can be evaluated as

$$\mathcal{L}(\boldsymbol{S}|\boldsymbol{\theta}) = \prod_{t=1}^{n} p_{\boldsymbol{\theta}}(S_t|\boldsymbol{S}_{1:t-1}),$$

so that an MCMC algorithm can be constructed directly on the parameter space (e.g., Andrieu et al., 2009, 2010).

For linear SDE models, all conditional distributions are Gaussian, leading to the celebrated Kalman filter (Kalman et al., 1960) with its analytic updates for latent variable means and variance matrices. For arbitrary nonlinear SDE models (such as the Heston

model), updates are performed by, so-called, particle filters (e.g., Gordon et al., 1993; Andrieu and Doucet, 2002; Andrieu et al., 2009, 2010), for which importance sampling estimators of $p_{\boldsymbol{\theta}}(V_t|\boldsymbol{S}_{1:t})$ have their weights updated by sequential Monte Carlo (SMC). However, particle filters impose a considerable computational burden, spurring an extensive literature on approximate filtering methods (e.g., Evensen, 1994; Andersen and Lund, 1997; Julier and Uhlmann, 1997; Song, 2000; Arasaratnam and Haykin, 2009), each striking its own balance between speed and accuracy.

In Chapter 4, we present a new filtering approximation termed Synthetic Kalman Filter (SKF), by analogy with the method of synthetic likelihoods of e.g., Wood (2010); Price et al. (2017). Leveraging the fact that the conditional densities of many mSV models are approximately Gaussian, a particle filter is used to update only mean and variance matrix, thus requiring far less particles than for the entire conditional distribution. An extensive simulation study indicates that our filter can approximate the filtering density and model likelihood accurately, at a fraction of the computational cost of the exact methods.

In Chapter 5, we summarize our contributions in the three projects in the aspects of computational gain, usage of the VIX, model frameworks proposed and inference approaches.

Chapter 2

Fast and Flexible Financial Modeling with an Observable Volatility Proxy

2.1 Introduction

SV models are important in financial modeling, since they capture the time-varying property of the volatility of a financial asset. The Black-Scholes model (Black and Scholes, 1973) assumes constant volatility, under which the options are often mispriced (Black, 1975), so that the early SV models (Hull and White, 1987; Scott, 1987; Wiggins, 1987) were proposed in an attempt to remedy this shortcoming. Typically, an SV model is given by two SDEs, one for the asset price denoted by S_t, and the other for the asset price volatility on the standard deviation scale, which is denoted by V_t. For example, a popular SV model is that of Heston (1993), which can be expressed as

$$\begin{aligned} \mathrm{d}X_t &= (\alpha - \tfrac{1}{2}V_t^2)\,\mathrm{d}t + V_t\,\mathrm{d}B_t^X, \\ \mathrm{d}V_t &= (\beta/V_t - \tfrac{1}{2}\gamma V_t)\,\mathrm{d}t + \tfrac{1}{2}\sigma\,\mathrm{d}B_t^V, \end{aligned} \qquad \mathrm{cor}(B_t^X, B_t^V) = \rho, \tag{2.1}$$

where $X_t = \log(S_t)$ is the log price. Compared to the other class of time-varying volatility models, i.e., the generalized autoregressive conditional heteroscedasticity (GARCH) models (Engle, 1982; Bollerslev, 1986), which assume that the volatility is determined by previous

states, the stochastic modeling fits to the inherent uncertainty of the volatility, so that SV models are favored theoretically (Fleming and Kirby, 2003). However, estimation of the parameters of SV models is considerably more difficult than for those of GARCH models for two reasons:

1. The data for estimating the SV model parameters is typically of the form $X_{0:n} = (X_0, ..., X_n)$, where X_t, $t = 0, \ldots, n$ is the observation at time $t \times \Delta\mathrm{T}$ and $\Delta\mathrm{T}$ is the inter-observation time. While the SDE underlying the SV model is a Markov process, its transition density $p(X_{t+1}, V_{t+1}|X_t, V_t, \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ represents the model parameters, is not available in closed-form.

2. The volatility process V_t is completely latent, such that the observed data likelihood is
$$\mathcal{L}(\boldsymbol{\theta}|X_{0:n}) = p(X_{0:n}|\boldsymbol{\theta}) = \int \prod_{i=1}^{n} p(X_i, V_i|X_{i-1}, V_{i-1}, \boldsymbol{\theta}) dV_{0:n}, \tag{2.2}$$
where $V_{0:n} = (V_0, ..., V_n)$ is defined similarly as $X_{0:n}$. Therefore, in order to estimate the model parameters, the unobserved $V_{0:n}$ must be integrated out from the likelihood.

The first problem is usually resolved with an approximate discretization scheme. For the Heston model this is
$$\begin{aligned} \Delta X_t &= (\alpha - \tfrac{1}{2}V_t^2)\Delta\mathrm{T} + V_t \Delta B_t^X \\ \Delta V_t &= (\beta/V_t - \tfrac{1}{2}\gamma V_t)\Delta\mathrm{T} + \tfrac{1}{2}\sigma \Delta B_t^V, \end{aligned} \qquad \begin{pmatrix} \Delta B_t^X \\ \Delta B_t^V \end{pmatrix} \overset{\text{iid}}{\sim} \mathcal{N}_2\left(\mathbf{0}, \Delta\mathrm{T}\begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right), \tag{2.3}$$
where $\Delta X_t = X_{t+1} - X_t$, $t = 0, \cdots, n-1$ (the notations with Δ in this chapter are defined similarly). The volatility is expressed on standard deviation scale instead of the variance scale. The variance process follows a CIR process. We favor this version since the volatility of the volatility becomes constant $(\sigma/2)$, in which case the Euler-Maruyama approximation has a faster rate of convergence (Mil'shtein, 1979). Then the density $p(X_t, V_t|X_{0:t-1}, V_{0:t-1}, \boldsymbol{\theta})$ follows a bivariate Gaussian distribution. Empirical evidence indicates that this approximation is generally usable for financial data recorded at daily intervals (Phillips and Yu, 2009). On the other hand, the problem of integrating over V_t, even in the approximate likelihood (2.2), must be handled by computationally intensive methods like the Markov chain Monte Carlo (MCMC), see Eraker (2001) for example.

Note that the volatility has to stay positive. For its continuous-time model, the volatility process stays positive as long as the parameters satisfy the condition of $\beta > \sigma^2/2$, which is derived from the CIR condition. By discretization, there exist slight chances that the volatility can be negative. We deal with such scenario by simply rejecting the simulations and posterior draws with negative volatilities.

While inference for a SV model with unobserved volatilities poses a considerable computation challenge, the problem is far simpler when the volatilities are observed. Indeed, for the discretized Heston model (2.1), $\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}|X_{0:n}, V_{0:n})$ can be obtained almost effortlessly via regression techniques. Moreover, though the volatilities of the model are unobserved, there is considerable information in financial markets by which to approximate them outside of a pure likelihood framework. Rogers and Satchell (1991); Alizadeh et al. (2002); Engle and Gallo (2006); Martens and Van Dijk (2007) use intraday information to estimate the daily volatility. Andersen and Bollerslev (1998); Andersen et al. (2003) employ high frequency return data to estimate volatility. Ledoit and Santa-Clara (1998) consider the Black-Scholes implied volatility calculated from the market traded at-the-money short-maturity options to be a rather good candidate proxy for instantaneous volatility. If the underlying asset is the SPX index, the Volatility Index (VIX) (Whaley, 1993) calculated by the Chicago Board Options Exchange (CBOE) is a popular measure of its volatility. Due to the easy access to the VIX data and the VIX itself is a volatility estimator needed no further parameter based calculation, we choose to proceed with it as the volatility estimator in this chapter.

The VIX is a market volatility index implied by the SPX at-the-money options. It is also known as the fear index (Whaley, 2009), in the sense that the higher VIX indicates a higher level of fear for a shock in the financial market. According to its definition by Whaley (1993), the value of the VIX is interpreted as the percentage of the expected annual change of the SPX with a 68% confidence level, and the volatility V_t in the discretized Heston model (2.3) represents a similar quantity. Since the VIX is on a percentage scale while the parameter σ is not, the value of the VIX should be around 100 times larger than the real underlying SPX volatility if the VIX mimics the underlying volatility perfectly.

To check the dynamics of the VIX and Heston volatility of the SPX, the adjusted daily closing price time series of SPX from 1990 to 2014 is used to estimate the Heston

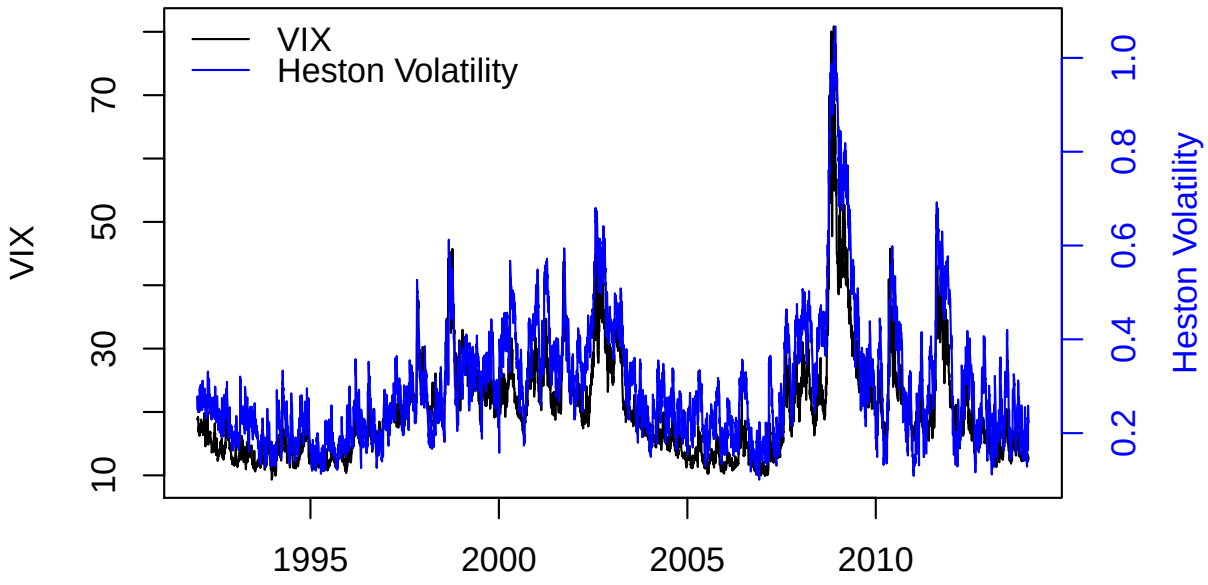


Figure 2.1: The VIX and the Heston volatility of the SPX.

model in a Bayesian framework with non-informative prior so that we can collect the posterior samples for the volatilities and the parameters. We extract the mean of the Heston volatility posterior samples to be a point estimator of the instantaneous volatility. Figure 2.1 shows the comparison. Although the two lines are not exactly the same by a ratio of 100, it is clear that their dynamics are very close. A similar result can also be found in Polson and Stroud (2003).

In this chapter, we will study the effect of substituting the VIX into the place of the volatilities in a SV framework to analyze the SPX from 1990 to 2014. To do this, we cast the fully observed SV structure into the framework of seemingly unrelated regressions (SUR) (Zellner, 1962). The SUR model is a more flexible framework than the conventional multivariate linear model, so that the inference of the SUR model is not as straightforward as conventional multivariate linear model. Feasible generalized least squares (Fomby et al., 1984) method is often used to estimate the SUR parameters. (Percy, 1992; Zellner and Ando, 2010; Ando et al., 2010) also discuss the SUR inference in a Bayesian context. We find that the Gibbs sampler introduced by Percy (1992) does not scale with the number

of the observations and the conditional draws are from analytical densities. Hence, it can produce reliable posterior draws using a smaller amount of time. In this chapter, our posterior analysis of the SUR models adopts this Gibbs sampler.

It is well-known that the financial data follows a distribution which is skewed (Kraus and Litzenberger, 1976) and has heavy tail (Mandelbrot, 1963). A non-Gaussian innovation distribution is a common way to deal with it. For example, Bollerslev (1987) presents the GARCH model with the t-distribution to capture the heavy tails of the distribution and finds it fits better with exchange rate data than the Gaussian innovation; moreover, Jacquier et al. (2004) provide a Bayesian inference algorithm for the simple SV model (1.4) with t-distributed innovation and Haas et al. (2004) find a mixture of Gaussian distribution to be another good candidate for the distribution of the innovation. Instead of proposing a non-Gaussian innovation, we provide a good fit by transforming the variable. We show how to do this with an empirical transformation which is compatible with the SUR framework, such that exact Bayesian inference and forecasting can still be carried out.

This chapter is organized as follows. In Section 2.2, we present the Bayesian method for the SUR model and the empirical transformation. In Section 2.3, we carry out an out-of-sample forecasting back-test, which is based on the Rosenblatt residual (Rosenblatt, 1952) results, for several SUR models of the SPX and the VIX. The Heston model and the GARCH$(1,1)$ model are used as benchmarks. Section 2.4 is the conclusion.

2.2 The Seemingly Unrelated Regression for SV-like Models

The multivariate linear (ML) model (Mardia and Kent, 1979) is a natural extension of the univariate linear model. It has the form

$$\boldsymbol{y}_t = \boldsymbol{x}_t'\boldsymbol{\beta} + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \overset{\text{iid}}{\sim} \mathcal{N}_q(\mathbf{0}, \boldsymbol{\Sigma}), \quad t = 1, \cdots, n$$

where $\boldsymbol{y}_t$ is a vector of q responses, $\boldsymbol{x}_t$ is a p dimensional covariate vector and $\boldsymbol{\beta}$ is a $p \times q$ coefficient parameter matrix. In the ML framework, each response y_{ti} shares the

same covariate $\boldsymbol{x}_t$ and the responses are connected by the correlated error vector $\boldsymbol{\epsilon}_t = (\epsilon_{t1}, \ldots, \epsilon_{tq})$. In contrast, the responses in the SUR model can have different covariates, such that

$$\begin{aligned} y_{t1} &= \boldsymbol{x}_{t1}'\boldsymbol{\beta}_1 + \epsilon_{t1} \\ \vdots \quad &\vdots \quad \vdots \qquad\qquad \boldsymbol{\epsilon}_t \overset{\text{iid}}{\sim} \mathcal{N}_q(\mathbf{0}, \boldsymbol{\Sigma}), \\ y_{tq} &= \boldsymbol{x}_{tq}'\boldsymbol{\beta}_q + \epsilon_{t1}, \end{aligned} \tag{2.4}$$

where we have q responses that each has their own covariate vector $\boldsymbol{x}_{ti}$ of dimension p_i and coefficient parameters $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_q)$.

To demonstrate the relevance of the SUR framework to SV modelings, consider the exponential Ornstein-Uhlenbeck (eOU) model of Scott (1987); Fouque et al. (2000), whose discretized form is given by

$$\begin{aligned} \Delta X_t &= (\alpha - \tfrac{1}{2}V_t^2)\Delta\mathrm{T} + V_t \Delta B_t^X, \\ \Delta \log V_t &= \gamma(\mu - \log V_t)\Delta\mathrm{T} + \sigma \Delta B_t^V, \end{aligned} \qquad \begin{pmatrix} \Delta B_t^X \\ \Delta B_t^V \end{pmatrix} \overset{\text{iid}}{\sim} \mathcal{N}_2\left(\mathbf{0}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right). \tag{2.5}$$

In this model, $X_t, \log V_t \in \mathbb{R}$ so that it avoids the restriction of the Heston model (2.3) that $V_t > 0$. Mainly for this numerical convenience, we choose the eOU model as the target model in this chapter. The eOU model can be cast in the SUR framework by setting

$$\begin{aligned} y_{t1} &= \Delta X_t / V_t & \boldsymbol{x}_{t1} &= (V_t^{-1}, V_t) \\ y_{t2} &= \Delta \log V_t & \boldsymbol{x}_{t2} &= (1, \log V_t), \end{aligned}$$

where we plug in the volatility V_t with the VIX. As it shown in Figure 2.1 and according to the definition of the VIX in Section 2.1, the latent volatility V_t is approximately proportional to the VIX. This ratio together with σ and ρ are represented by the variance-covariance matrix $\boldsymbol{\Sigma}$ in (2.4). We also relax the coefficient of V_t^2 in the first equation of (2.5) to be a model parameter. Together with the other coefficient parameters in (2.5), they form the $\boldsymbol{\beta}$ in (2.4).

More generally, the SUR framework encompasses a wide range of nonlinear autoregressive time series models. Denote the ith underlying time-series observations from time 0

to time t by $\boldsymbol{z}_{0:t,i} = (z_{0,i}, \ldots, z_{t,i})$ and $\boldsymbol{z}_{0:t} = (\boldsymbol{z}_{0:t,1}, \ldots, \boldsymbol{z}_{0:t,q})$, then a generic nonlinear autoregressive model

$$g_i(z_{t,i}, \boldsymbol{z}_{0:t-1}) = \sum_{j=1}^{p_i} \beta_{ij} f_{t,j}(\boldsymbol{z}_{0:t-1}) + h_i(\boldsymbol{z}_{0:t-1})\epsilon_{t,i}, \quad \boldsymbol{\epsilon}_t = [\epsilon_{t,i}]_{i=1,\ldots,q} \overset{\text{iid}}{\sim} \mathcal{N}_q(\mathbf{0}, \boldsymbol{\Sigma}), \quad (2.6)$$

where $\boldsymbol{\beta}_i = (\beta_{i1}, \ldots, \beta_{ip_i})'$, can be rewritten into the form of the SUR model by setting

$$\begin{aligned} y_{t,i} &= g_i(z_{t,i}, \boldsymbol{z}_{0:t-1})/h_i(\boldsymbol{z}_{0:t-1}), \\ \boldsymbol{x}_{ti} &= [f_{t,1}(\boldsymbol{z}_{0:t-1})/h_i(\boldsymbol{z}_{0:t-1}), \ldots, f_{t,p_i}(\boldsymbol{z}_{0:t-1})/h_i(\boldsymbol{z}_{0:t-1})]', \end{aligned} \quad (2.7)$$

where $g_i(\cdot)$, $f_{t,i}(\cdot)$ and $h_i(\cdot)$ are functions to be specified. For example, casting the eOU model (2.5) in this generic form (2.6), the log-price process is $\boldsymbol{z}_{0:n,1}$, the VIX process is $\boldsymbol{z}_{0:n,2}$. For the unique prediction purpose, it is mandatory that $g_i(\cdot)$ is an invertible function with respect to the first argument $z_{i,t}$. In other words, there should exists $z_{t,i} = g_i^{-1}(y_{t,i}, \boldsymbol{z}_{0:t-1})$, which is usually satisfied easily.

The SUR model is very flexible with multivariate time-series modeling. The requirements for the functions are not strict, so the SUR model can be highly nonlinear with respect to $\boldsymbol{z}_{0:n}$. We can model as many time series together as the problem requires, which is suitable to the multi-asset models. The number of the covariates can be different for different response. In conclusion, the SUR model includes a huge collection of multivariate nonlinear autoregressive models and it can be applied to various problems, certainly not limited to the financial modeling.

2.2.1 Bayesian Inference

The inference for the SUR model is not the same as the ML model. The least square estimator or maximum likelihood estimator for an ML model is very easy to acquire. Although there exist some least square methods for the SUR model (Srivastava and Dwivedi, 1979), but they are not straightforward to be applied. In this section, we show a Gibbs sampler for the SUR model which has been investigated by Percy (1992); Zellner and Ando (2010), and further discuss how to speed up the algorithm by pre-calculating some sufficient statistics.

The parameters in the SUR model include the coefficient parameters $\boldsymbol{\beta}$ and the variance-covariance matrix $\boldsymbol{\Sigma}$. For simplicity, we define $\boldsymbol{\theta} = (\boldsymbol{\beta}, \boldsymbol{\Sigma})$. Now denote $\boldsymbol{z}_t$ to be the vector of the q responses at time t, and $\boldsymbol{z}_{0:n} = (\boldsymbol{z}_0, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_n)$. Since the functions $g_i(\cdot)$, $i = 1, \ldots, q$, are invertible, $\boldsymbol{z}_{0:n}$ shares the same information as the responses $\boldsymbol{Y}$. Without loss of generality, we consider $\boldsymbol{y}_t = \boldsymbol{z}_t, t = 0, \ldots, n$ in the following context.

Note that $p(\boldsymbol{z}_t|\boldsymbol{z}_{0:t-1}, \boldsymbol{\theta}) = p(\boldsymbol{y}_t|\boldsymbol{x}_{t1}, \ldots, \boldsymbol{x}_{tq}, \boldsymbol{\theta})$ is a multivariate Gaussian density, so the model posterior density can be analytically obtained by

$$p(\boldsymbol{\theta}|\boldsymbol{Y}, \boldsymbol{X}) = p(\boldsymbol{\theta}|\boldsymbol{z}_{0:n}) \propto \pi(\boldsymbol{\theta}) p(\boldsymbol{z}_{0:n}|\boldsymbol{\theta}) = \pi(\boldsymbol{\theta}) \prod_{t=1}^{n} p(\boldsymbol{z}_i|\boldsymbol{z}_{0:t-1}, \boldsymbol{\theta}),$$

where $\pi(\boldsymbol{\theta})$ represents the parameters prior density. Then the log-posterior of the SUR model is

$$\begin{aligned} p(\boldsymbol{\theta}|\boldsymbol{Y}, \boldsymbol{X}) = & \log \pi(\boldsymbol{\theta}) - \frac{n}{2} \log |\boldsymbol{\Sigma}| \\ & - \frac{1}{2} \sum_{t=1}^{n} \Big(\boldsymbol{y}_t - \big(\boldsymbol{x}'_{t1}\boldsymbol{\beta}_1, \ldots, \boldsymbol{x}'_{tq}\boldsymbol{\beta}_q \big) \Big)' \boldsymbol{\Sigma}^{-1} \Big(\boldsymbol{y}_t - \big(\boldsymbol{x}'_{t1}\boldsymbol{\beta}_1, \ldots, \boldsymbol{x}'_{tq}\boldsymbol{\beta}_q \big) \Big). \end{aligned}$$

In the SUR framework, we can find a conjugate prior for $\boldsymbol{\theta}$,

$$\begin{aligned} \boldsymbol{\Sigma} &\sim \mathcal{W}_q^{-1}(\boldsymbol{\Psi}, \nu) \\ \boldsymbol{\beta} &\sim \mathcal{N}_d(\boldsymbol{\lambda}, \boldsymbol{\Omega}), \end{aligned}$$

where $d = \sum_{i=1}^{q} p_i$, with parameters $(\boldsymbol{\Psi}, \nu, \boldsymbol{\lambda}, \boldsymbol{\Omega})$. In Appendix A, we show the posterior log-density of $\boldsymbol{\theta}$ has the form

$$\begin{aligned} p(\boldsymbol{\theta}|\boldsymbol{Y}, \boldsymbol{X}) = & - \frac{n + \nu + q + 1}{2} \log |\boldsymbol{\Sigma}| - \frac{1}{2} \mathrm{Tr}(\boldsymbol{\Psi}\boldsymbol{\Sigma}^{-1}) - \frac{1}{2} (\boldsymbol{\beta} - \boldsymbol{\lambda})' \boldsymbol{\Omega}^{-1} (\boldsymbol{\beta} - \boldsymbol{\lambda}) \\ & - \frac{1}{2} \sum_{t=1}^{n} \Big(\boldsymbol{y}_t - \big(\boldsymbol{x}'_{t1}\boldsymbol{\beta}_1, \ldots, \boldsymbol{x}'_{tq}\boldsymbol{\beta}_q \big) \Big)' \boldsymbol{\Sigma}^{-1} \Big(\boldsymbol{y}_t - \big(\boldsymbol{x}'_{t1}\boldsymbol{\beta}_1, \ldots, \boldsymbol{x}'_{tq}\boldsymbol{\beta}_q \big) \Big), \end{aligned}$$

and the conditional posterior distributions of $\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}$ respectively are

$$\begin{aligned} \boldsymbol{\Sigma}|\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y} &\sim \mathcal{W}_q^{-1}(\boldsymbol{\Psi} + \boldsymbol{D}, \boldsymbol{N}), \\ \boldsymbol{\beta}|\boldsymbol{\Sigma}, \boldsymbol{X}, \boldsymbol{Y} &\sim \mathcal{N}(\boldsymbol{B}\boldsymbol{\lambda} + (I - \boldsymbol{B})\hat{\boldsymbol{\beta}}, (\boldsymbol{\Omega}^{-1} + \boldsymbol{V}^{-1})^{-1}), \end{aligned} \tag{2.8}$$

where $\boldsymbol{B} = (\boldsymbol{\Omega}^{-1} + \boldsymbol{V}^{-1})^{-1}\boldsymbol{\Omega}^{-1}$ and the analytic forms of the other sufficient statistics $\boldsymbol{D}$, $\boldsymbol{N}$, $\hat{\boldsymbol{\beta}}$ and $\boldsymbol{V}$ can be found in Appendix A. Based on those conditional densities, Percy (1992) builds a Gibbs sampler to obtain posterior samples by sampling back and forth between $\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}$.

If we choose a flat prior of

$$\begin{aligned} \pi(\boldsymbol{\beta}) &\propto 1, \\ \pi(\boldsymbol{\Sigma}|\boldsymbol{\beta}) &\propto |\boldsymbol{\Sigma}|^{(q+1)/2}, \end{aligned} \tag{2.9}$$

we can obtain the corresponding Gibbs sampler by setting all the prior parameters $(\boldsymbol{\Psi}, \nu, \boldsymbol{\lambda}, \boldsymbol{\Omega})$ to $\mathbf{0}$ so that

$$\begin{aligned} \boldsymbol{\Sigma}|\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{Y} &\sim \mathcal{W}_q^{-1}(\boldsymbol{D}, n), \\ \boldsymbol{\beta}|\boldsymbol{\Sigma}, \boldsymbol{X}, \boldsymbol{Y} &\sim \mathcal{N}(\hat{\boldsymbol{\beta}}, \boldsymbol{V}). \end{aligned}$$

To further illustrate the structure and discuss the computational order of this algorithm, we introduce more notations:

$$\begin{aligned} \boldsymbol{y}_t &= (y_{t1}, \ldots, y_{tq}), & \boldsymbol{Y} &= \left[y_{ti}\right]_{t=1,\ldots,n}^{i=1,\ldots,q}, & \boldsymbol{y}_{(i)} &= (y_{1i}, \cdots, y_{ni})' \\ \boldsymbol{x}_{ti} &= (x_{t1}^{(i)}, \ldots, x_{tp_i}^{(i)}), & \boldsymbol{X}_i &= \left[x_{tj}^{(i)}\right]_{t=1,\ldots,n}^{j=1,\ldots,p_i}, & \boldsymbol{X} &= (\boldsymbol{X}_1, \ldots, \boldsymbol{X}_q) \end{aligned}$$

Taking a closer look at the formulas for the sufficient statistics in Appendix A, we find that the heavy computations are mainly due to calculating $\boldsymbol{D}$ and $\hat{\boldsymbol{\beta}}$. $\boldsymbol{D}$ is a $q \times q$ matrix that the element of row i and column j is

$$\boldsymbol{D}_{i,j} = \left[(\boldsymbol{y}_{(i)}'\boldsymbol{y}_{(j)}) - \boldsymbol{\beta}_i'(\boldsymbol{X}_i'\boldsymbol{y}_{(j)}) - (\boldsymbol{y}_{(i)}'\boldsymbol{X}_j)\boldsymbol{\beta}_j + \boldsymbol{\beta}_i'(\boldsymbol{X}_i'\boldsymbol{X}_j)\boldsymbol{\beta}_j)\right]$$

where $i, j = 1, \cdots, q$ and $\boldsymbol{y}_{(i)}$ is the ith column of $\boldsymbol{Y}$. Note that the values $\boldsymbol{y}_{(i)}'\boldsymbol{y}_{(j)}$, $\boldsymbol{X}_i'\boldsymbol{y}_{(j)}$, $\boldsymbol{y}_{(i)}'\boldsymbol{X}_j$ and $\boldsymbol{X}_i'\boldsymbol{X}_j$ are costly to calculate, but this calculation is the same for every updating, so they can be pre-calculated before the Gibbs algorithm. As for $\hat{\boldsymbol{\beta}}$, it is straightforward that the parts concerning $\boldsymbol{X}$ and $\boldsymbol{Y}$ can also be pre-calculated (see the details in Appendix A).

The fact that these statistics can be pre-calculated is very important in the sense that during the iterative updates of the Gibbs sampler, we can avoid calculating the most

computationally demanding statistics repeatedly. The computational order of the pre-calculated $\boldsymbol{Y}$ and $\boldsymbol{X}$ related statistics is $\mathcal{O}(nd^2)$. Let us define $p = \max(p_i)$, then the computational order for each $\boldsymbol{\Sigma}$ updating step is $\mathcal{O}(\max(q^2p^2, q^3))$ and the computational order for each $\boldsymbol{\beta}$ updating is $\mathcal{O}(d^3)$ given those pre-calculated statistics. Hence, in each updating step, the computational order does not concern n, which is the length of the observations. The only step that n involved is during the pre-calculation of $\boldsymbol{Y}'\boldsymbol{Y}$, $\boldsymbol{Y}'\boldsymbol{X}$ and $\boldsymbol{X}'\boldsymbol{X}$, and indeed we only calculate it once. Basically, the computational order of the whole Gibbs sampler algorithm is determined mainly by the length of $\boldsymbol{\beta}$ (d), and the number of covariates (p). The details of the computational order can be checked in Appendix A.

2.2.2 Empirical Transformation

In the SUR framework (2.4), the error vector follows a multivariate Gaussian distribution. As we mentioned in the introduction, the financial assets usually do not follow the Gaussian distribution but are skewed and heavy-tailed. The obvious solution to obtaining a better fit is to replace the Gaussian errors with those from another distribution. However, we can easily break the fast Gibbs sampler described in the last subsection if we adopt a non-Gaussian distribution. In order to keep the exact analysis and computational efficiency, we perform an empirical transformation on the time-series before we fit the SUR model.

To illustrate this empirical transformation, we transform the first response $\Delta X_t/V_t$ in the discretized eOU model (2.5) as an example. Assume that the training dataset is $z_{i,1} = \Delta X_i/V_i$ and $z_{i,2} = \log(V_{i,1})$, $i = 0, \ldots, t$. Let us define

$$y_{i,1} = \Phi^{-1}(\mathrm{F}(z_{i,1})), \qquad i = 0, \ldots, t, \tag{2.10}$$

where $\mathrm{F}(\cdot)$ is the CDF of the empirical distribution constructed by $[z_{i,1}]_{i=1,\ldots,t}$ and $\Phi(\cdot)$ is the CDF of the best fitting Gaussian distribution, which uses the sample mean and variance of $[z_{i,1}]_{i=1,\ldots,t}$, so that $\Phi^{-1}(\cdot)$ is the corresponding quantile function. Indeed, this transformation is the $g_1(\cdot)$ function in (2.7). In this way, we first find the probabilities of $[z_{i,1}]_{i=1,\ldots,t}$ in their potentially skewed and heavy-tailed empirical distribution and then project them to the best-fit Gaussian quantiles $[y_{i,1}]_{i=1,\ldots,t}$. Subsequently, when we fit this

new time-series $[y_{i,1}]_{i=1,\dots,t}$ in the SUR model, we shall expect to have a better fit since the transformed training data fits better to the Gaussian innovations. When we attempt to predict $z_{t+1,1}$, an inverse transformation of (2.10), which consists of functions $\Phi(\cdot)$ and $\mathrm{F}^{-1}(\cdot)$, shall be applied to every $y_{t+1,1}$ prediction with the same underlying Gaussian and empirical distribution of (2.10).

The CDF of a Gaussian distribution is simple and clear, but there exist several approaches to find the empirical quantile. In practice, we use the Gaussian kernel density with bandwidth chosen by the *rule of thumb* of Silverman (2018) to fit the empirical distribution. One evaluation of the corresponding CDF and quantile function are both in the order of $\mathcal{O}(n)$. Hence, this empirical transformation step takes $\mathcal{O}(n)$ to prepare the variable from $z_{t,1}$ to $y_{t,1}$ and the computational order is significantly lower than the Gibbs sampler (2.8). The empirical transformation step should not affect the total computational time of the inference significantly.

The empirical distribution of the training dataset $[z_{i,1}]_{i=1,\dots,t}$ can be viewed as an approximation for their stationary distribution, about which we obtain more and more information as the time t grows. If the size of the observations is too small, the accuracy of the approximation is typically low. Hence, we set a minimum size of the training dataset to be 500. Figure 2.2 shows the time series $z_{t,1} = \Delta X_t / V_t$. First, we can see a clear heavier tail on the negative side, which indicating the data we provide without this transformation to the Gaussian SUR model is not Gaussian distributed. Second, the range of the empirical density changes rarely. Two red lines mark how the bounds of the empirical distribution change. The minimum size of 500 is marked by the black vertical line. Change of the red bounds means the range of prediction distribution until the previous day fails to cover the current day observation, which might be a problem to the prediction accuracy. However, the bounds only change 6 times in the long period of 5500 days after the black vertical line, we believe that the effects on the summary results are very limited.

This approach has the following advantages. First, the model likelihood is still based on Gaussian density, which keeps the algorithm straightforward and the inference exact. The complicated density of some heavy tail and skewed innovation distributions can easily increase the computational order of the algorithm and make the exact posterior density inference impossible. Second, the prediction distribution will have the skewness and heavy

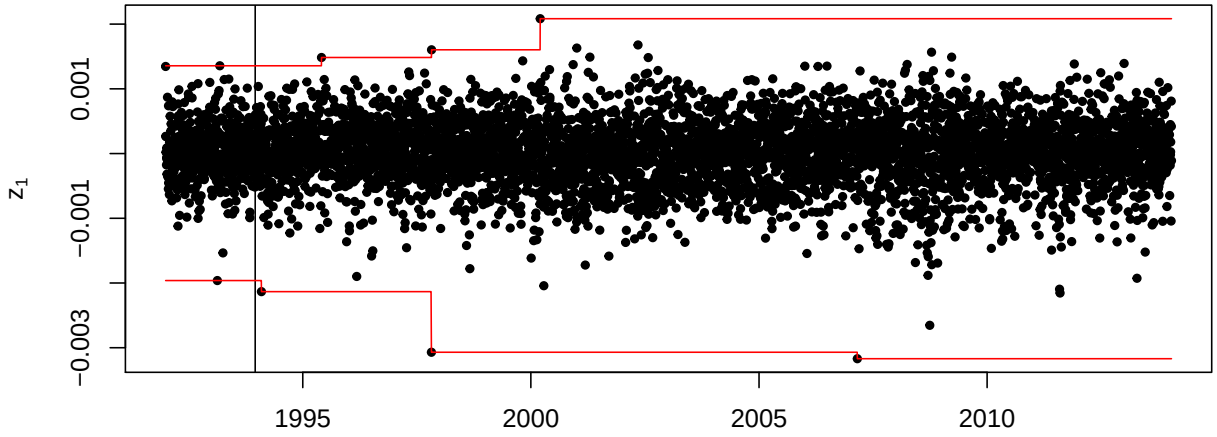


Figure 2.2: Ranges of $z_{t,1} = \Delta X_t / V_t$. Black line marks the minimum size of the training dataset.

tail properties as close to the empirical behavior of the returns as possible.

2.3 Goodness-of-Fit Assessment

In the following section, we shall compare the forecasting performance of our SUR model to other time-dependent volatility alternatives, namely the Heston and GARCH(1,1) models. To do this, we describe a powerful Bayesian out-of-sample testing strategy based on the Rosenblatt residuals (Rosenblatt, 1952).

2.3.1 Bayesian Rosenblatt Residuals for Multivariate Time Series

Consider n observations of a q-dimensional time series $\boldsymbol{z} = (\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n)$. Now suppose we wish to test the hypothesis that these data come from a given distribution:

$$H_0 : \boldsymbol{z} \sim p_0(\boldsymbol{z}).$$

Suppose $T_t = g(\boldsymbol{z}_t)$ is a given test statistic, with conditional CDF under H_0 given by

$$F_0(\tau|\boldsymbol{z}_{1:t-1}) = \text{Prob}_0(T_t < \tau|\boldsymbol{z}_{1:t-1}),$$

and let $U_t = F_0(T_t|\boldsymbol{z}_{1:t-1})$. Then we have

$$H_0 : U_t \overset{\text{iid}}{\sim} \text{Unif}(0,1), \qquad t = 1, \ldots, n.$$

Thus, the Rosenblatt residuals $U_1, ..., U_n$ leverage the power of n i.i.d. one-step forecasting tests to provide evidence against H_0. Similarly, we can calculate the k-step Rosenblatt residuals

$$U_{t,k} = F_{0,k}(T_{t+k}|\boldsymbol{z}_{1:t-1}), \qquad F_{0,k}(\tau|\boldsymbol{z}_{1:t-1}) = \text{Prob}_0(T_{t+k} < \tau|\boldsymbol{z}_{1:t-1}).$$

Under H_0, these residuals are no longer independent, but still each Unif$(0,1)$.

In practice, often we wish to assess the goodness-of-fit of a parametric family of models, $\boldsymbol{z} \sim p(\boldsymbol{z}|\boldsymbol{\theta})$. In a Bayesian context, we can test against a reference distribution of the form

$$H_0 : \boldsymbol{z} \sim p(\boldsymbol{z}|\boldsymbol{z}_{\text{train}}) = \int p(\boldsymbol{z}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{z}_{\text{train}})d\boldsymbol{\theta},$$

where $\boldsymbol{z}_{\text{train}} = (z_{-m}, z_{-m+1}, \ldots, z_0)$ is a set of time series observations preceding $\boldsymbol{z}$, and the posterior on the training data $p(\theta|\boldsymbol{z}_{\text{train}})$ becomes the prior for the testing data $\boldsymbol{z}$. Calculating the Bayesian Rosenblatt residuals requires the conditional CDFs of $p(T_{t+k}|\boldsymbol{z}_{1:t-1}, \boldsymbol{z}_{\text{train}})$, which typically are analytically intractable. Algorithm 2.1 shows how to estimate U_{t+k} by Monte Carlo simulation.

We choose the daily adjusted closing prices of the SPX and the VIX from 1990 to 2014 as the data being investigated. There are over 6000 observations in this dataset, which allows us to iterate Algorithm 2.1 for 5500 times. In other words, we can collect 5500 sets of 1-step and 10-step predictive distributions and the corresponding u_{t+k}. The objective of this exercise is to compare the out-of-sample prediction power of the SUR model to that of the other benchmark models using the experimental results by using Algorithm 2.1. Unfortunately, the three models cannot be estimated by the same algorithm. The SUR model is fitted by the proposed Gibbs sampler (2.8) using a non-informative prior (2.9). The benchmark models include the GARCH$(1,1)$ model and the Heston model. The R

Algorithm 2.1: SUR model experiment steps

1 Set $m = -499$, so that $\boldsymbol{z}_{\text{train}}$ is of length 500.

2 Start the algorithm with $t = 1$ and the training dataset is $(\boldsymbol{z}_{1:t-1}, \boldsymbol{z}_{\text{train}})$, where $\boldsymbol{z}_{1:t-1}$ includes no data when $t = 1$.

3 Draw a posterior sample of size $M = 100,000$ as $\boldsymbol{\theta}_t^{(j)} \sim p(\boldsymbol{\theta}|\boldsymbol{z}_{\text{train}}, \boldsymbol{z}_{1:t-1}), j = 1, \ldots, M$ by the SUR model Gibbs sampler as described in Section 2.2.1.

4 Simulate 10 steps forward by $\boldsymbol{z}_{t+l}^{(j)} \sim p(\boldsymbol{z}_{t+l}|\boldsymbol{z}_{t+l-1}^{(j)}, \boldsymbol{\theta}_t^{(j)})$, $l = 1, 2, \ldots, 10$ for every $\boldsymbol{\theta}_t^{(j)}$ and collect 1-step forecasting $\left[\boldsymbol{z}_{t+1}^{(j)}\right]_{j=1,\ldots,M}$ and 10-step forecasting $\left[\boldsymbol{z}_{t+10}^{(j)}\right]_{j=1,\ldots,M}$.

5 Calculate the Monte Carlo estimators of U_{t+k} as

$$u_{t+k} = \frac{1}{M}\sum_{j=1}^{M} I_{\boldsymbol{z}_{t+k}^{(j)} < \boldsymbol{z}_{t+k}}, \quad k = 1, 10.$$

6 Replace t by $t + 1$, so that a new data point $\boldsymbol{z}_t$ is incorporated into the training data set of $(\boldsymbol{z}_{1:t-1}, \boldsymbol{z}_{\text{train}})$ then repeat step 3 to 5.

7 Loop step 6 until the end of the dataset.

package *rugarch* (Ghalanos, 2018) is used to fit GARCH(1, 1) model on the SPX data. The R package *msde* (Lysy and Tong, 2017) is use to fit the Heston model on the SPX data with a non-informative prior, which is detailed in Appendix B. Therefore, for the GARCH(1, 1) and Heston model, Step 3 of Algorithm 2.1 need to be changed into their corresponding parameter estimation schemes. As long as the parameter estimation algorithms converge, the forecasting results should represent the model's performance with the same observable

data.

2.3.2 Experiment for the Simple SUR Models

By substituting the VIX, the eOU model (2.5) can be expressed with a simple structure in the SUR framework. Since the high computational requirement is no longer an issue, we can actually afford to investigate other models proxied by the VIX in details. We anticipate that the eOU model has a limited power to describe several features of the data, for example the long term memory of the asset return. To show the flexibility of the SUR model, we introduce the following generalized model,

$$\begin{aligned} \Delta X_{t+1} &= \beta_{11} + \beta_{12} V_t^{\lambda_1} + \sum_{i=0}^{l} \beta_{1,3+i} X_{t-i} + \sum_{i=0}^{l} \beta_{1,4+l+i} V_{t-i} + V_t^{\lambda_2} \epsilon_{t1} \\ \Delta \log V_{t+1} &= V_t^{\lambda_3} (\beta_{21} - \beta_{22} \log(V_t)) + V_t^{\lambda_4} \epsilon_{t2} \end{aligned} \tag{2.11}$$

where $\epsilon_t = (\epsilon_{t1}, \epsilon_{t2})' \overset{\text{iid}}{\sim} \mathcal{N}_2(\mathbf{0}, \boldsymbol{\Sigma})$, V_t represents the VIX and the tuning parameters $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \lambda_3, \lambda_4)$ and l, which are predetermined. In fact, the discretized versions of many existing models, e.g., the eOU model and the Heston model, belong to this generalized model with the suitable selections of λ and l. It is possible to let the data determine $\boldsymbol{\lambda}$ by adding an extra step in the Gibbs sampler of drawing λ given $\boldsymbol{\beta}$, $\boldsymbol{\Sigma}$ and the data. However, this means the statistics $\boldsymbol{Y'Y}$, $\boldsymbol{Y'X}$ and $\boldsymbol{X'X}$ change in each update step of the Gibbs sampler, so that they needed to be recalculated in each step with a computational order of $\mathcal{O}(nd^2)$. It breaks the originally fast algorithm. Hence, we select several candidate sets of (l, λ) and experiment on them with their forecasting results and their Rosenblatt residuals. We find that $l = 2$ and $\boldsymbol{\lambda} = (0.75, 1, 0.5, 0.5)$ are a good set of choice. Hence, we decide to use the model with the above set of parameters, which we refer to as the Gen-VIX model, as an example of this group of generalized models and compare its results to the benchmark models in the following analysis part. Instead, we consider the Gen-VIX model as being an ideal candidate from within the family of the generalized models (2.11).

Figure 2.3 is a plot of the means and the 95% prediction intervals (PIs) of the 1-step prediction. The competing models include the eOU model with the VIX proxy (eOU-VIX), the Gen-VIX model, the Heston model, the GARCH$(1,1)$ model with the Gaussian

innovation and the GARCH$(1,1)$ with the t-distributed innovation in which the degree of freedom of the t-distributed is a model parameter adapted to different training data sets. We center the means, the PIs and the true observations by the means of the eOU-VIX model predictions so that we can inspect the intervals easier. The eOU-VIX model seems not to be sensitive enough judging by the relatively narrower PIs. The predictions of the Heston model show much more flexibility. Through assigning a specific value of λ, the Gen-VIX model obtains similar fluctuations as the Heston model. If we treat the Heston model as the benchmark here, it seems that the the PIs of the Gen-VIX model are relatively closer to the Heston model, while the PIs of the two GARCH models are too wide generally.

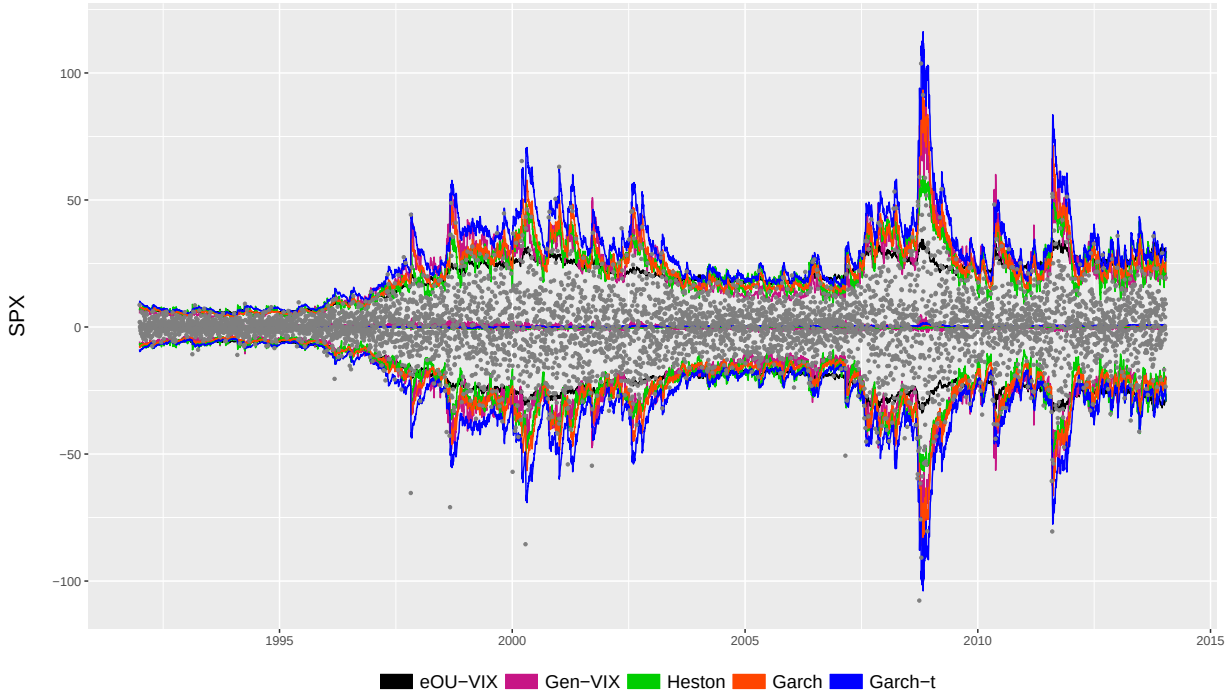


Figure 2.3: 1-step prediction for SPX price. Grey dots are the true observations. All of the predictions and the observations are centered by the mean of the eOU-VIX prediction.

We refer to the observations outside the 95% PIs as the outside values (OV) and the percentage of the OV among the 5500 observations can be seen in Table 2.1. There are

consequences of having narrower PIs, the eOU-VIX model has the largest percentage, which deviates from the 5% nominal value. On the other hand, the GARCH model with t-distributed innovation has too wider PIs, so that it has the lowest percentage which also deviated from the benchmark. Although the PIs of the Gen-VIX model seems to cover a quite similar range as the PIs of the Heston model, the results in the table show that the Gen-VIX model is not as good as the Heston model. This might due to that the PIs of the Gen-VIX model occasionally can be quite narrow, especially during the period of 2007 to 2008.

	eOU-VIX	Gen-VIX	Heston	GARCH	GARCH-t
1-step Pred.	7.92 (0.36)	6.32 (0.33)	5.81 (0.32)	6.16 (0.32)	2.77 (0.22)
10-step Pred.	13.8 (0.47)	12.6 (0.45)	3.13 (0.24)	3.28 (0.24)	1.37 (0.16)

Table 2.1: Summary of the percentage of the OV.

The longer term prediction is also an interesting topic in financial modeling. In Figure 2.4, we compare the 10-step prediction intervals among those models. Unlike the 1-step prediction, the Heston model seems to have clearly wider PIs than the Gen-VIX model in the 10-step prediction. The PIs of the eOU-VIX model are constantly narrower than the PIs of the other models. The 10-step prediction amplifies the fact that the PIs of the eOU-VIX model are slightly narrower than the others in 1-step prediction. Table 2.1 also reports the percentages of the corresponding OV of 10-step forecasting, and we see that all of the five percentages are not very close to 5%. Our proposed models and the GARCH model with t-distributed innovations are very far from the benchmark.

After the comparisons of the PIs, we check the null hypothesis of the u_{t+k} being Unif$(0, 1)$ according to the Rosenblatt residuals theory. As described in the Algorithm 2.1, we record the probabilities $[u_{t+k}]^{k=1,10}_{t=1,\ldots,5500}$ of the observation within the simulated predictive distribution. In the following, we test those probabilities are from Unif$(0, 1)$ distribution and the 1-step forecasting Rosenblatt residuals are independent.

Figure 2.5 shows the q-q plot and the density plot of the normalized $[u_{t+k}]^{k=1,10}_{t=1,\ldots,5500}$. That is, we transform those probabilities using a standard Gaussian quantile function, such that the transformed values should follow a standard Gaussian distribution. We provide

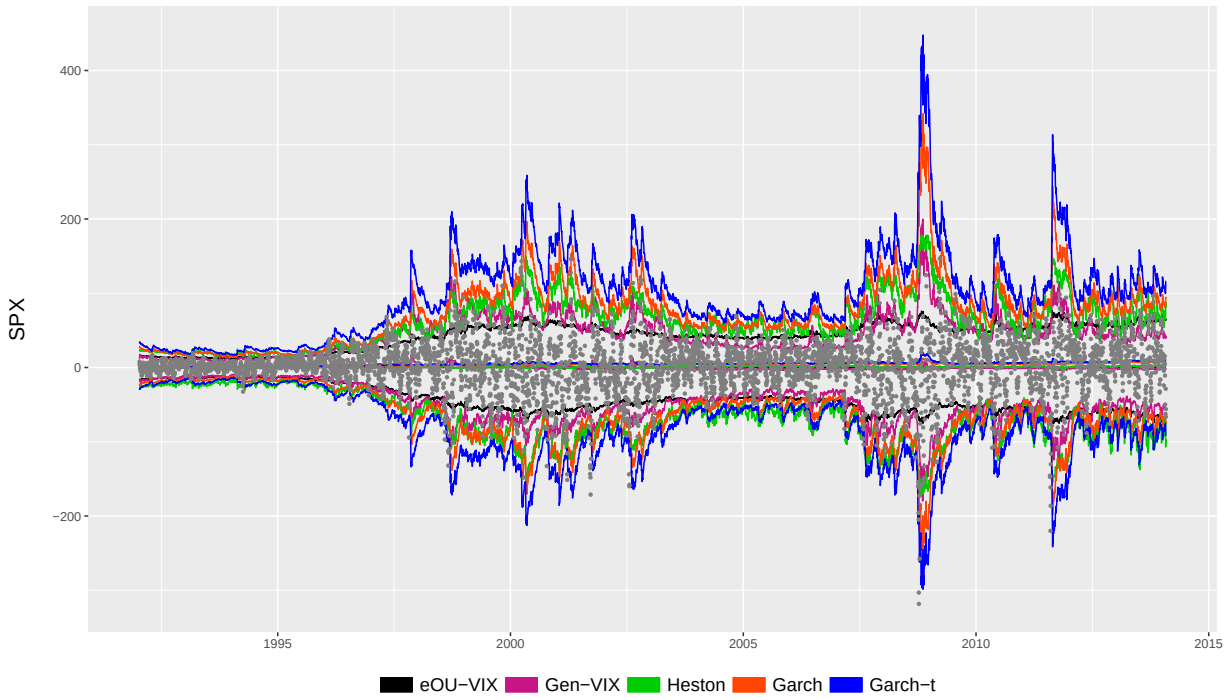
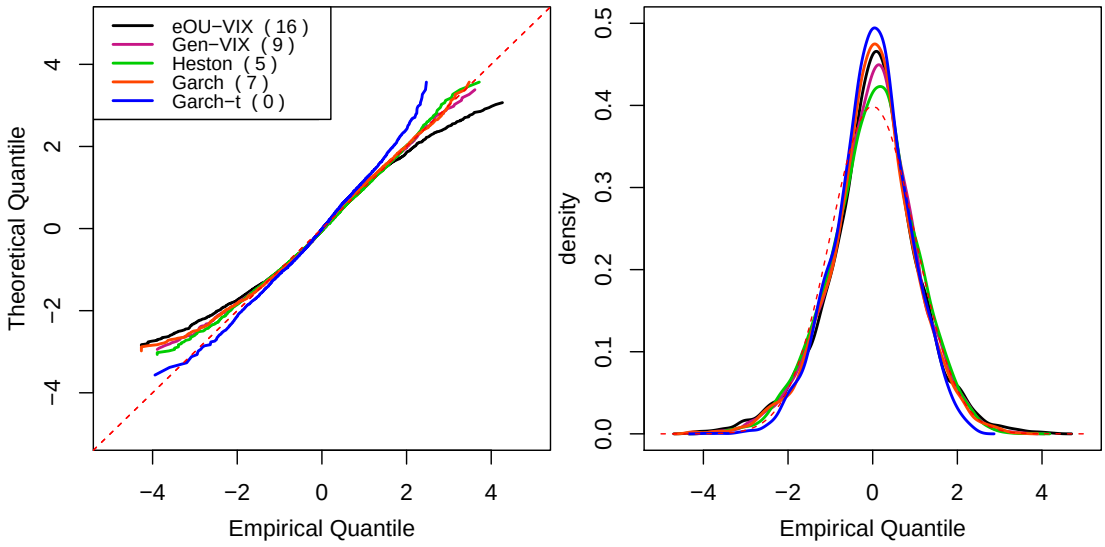
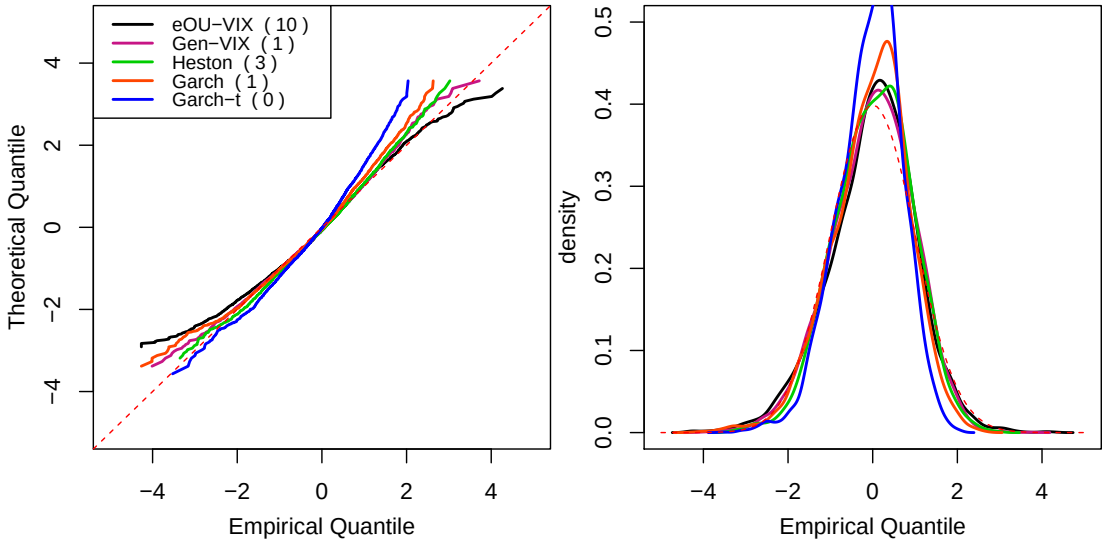


Figure 2.4: 10-step prediction for SPX price. Grey dots are the true observations. All of the predictions and the observations are centered by the mean of the eOU-VIX prediction.

the benchmark lines in red so that the skewness and the tail behaviors can be seen. In the 1-step prediction, the GARCH$(1,1)$-t model has an almost perfect negative tail but its positive tail deviates far from the benchmark. In general, the Heston model is slightly better than the others in the 1-step prediction. For the 10-step prediction, the curves of the Gen-VIX model and the Heston model are closer to the benchmark lines than those of the other 3 models. Specifically, the Heston model is better in the negative tail but the Gen-VIX model is better in the positive tail. The number in the legend shows the number of observations outside the support of the simulated forecasting distributions. Ideally, the number is close to 0. The GARCH-t model usually has the widest PIs, hence, it has no observation outside the empirical intervals. Our eOU-VIX and Gen-VIX model have the largest numbers. Even for the Heston model, the ranges of its simulated forecasting distributions cannot cover all the observations, which means the extreme cases do happen.



(a) P-values of 1-step predictions



(b) P-values of 10-step predictions

Figure 2.5: P-value diagnostic plots. Red dashed lines are the bench marks. The numbers in the legend shows the number of observations lie outside the supports of their empirical predictive distributions.

Note that our training data covers the year 2008, which is the year of the financial crisis, it is understandable that these numbers are not 0. The GARCH-t model seems to be advantageous of protecting us from the extreme situations like the financial crisis, however, this result is obtained at a cost of not able to predict as well as the other models during the ordinary period.

	eOU-VIX	Gen-VIX	Heston model	GARCH(1,1)	GARCH(1,1)-t
1-step pred.	0.068	0.004	0.101	0.068	0.059

Table 2.2: Summary of the p-values from the 1-step Ljung-Box test. GARCH(1,1) represents the GARCH(1,1) model with Gaussian innovation and GARCH(1,1)-t represents the GARCH(1,1) model with t-distributed innovation.

We have inspected $[u_{t+k}]_{t=1,\ldots,5500}^{k=1,10}$ as samples from Unif$(0,1)$ distribution in Figure 2.5, we should also check the assumption of independence for 1-step prediction. The possible scenario of OV clustering is a clear violation of this assumption and the ideal situation is that those OVs distributed uniformly across the whole time line. The Ljung-Box test is performed to test the null hypothesis that the p-values are independent at one lag and the results are shown in the Table 2.2. For the 1-step prediction, the Gen-VIX model rejects the null hypothesis at a significant level of 1% and the Heston model has the highest test p-value. By this result, we believe the Rosenblatt residuals of the Heston model is less autocorrelated at one lag.

In summary, the Heston model with the latent volatility requires the longest computational time, however, it provides better results than the other models in the 1-step prediction experiment. That is, it has the best OV percentage as shown in Table 2.1, the highest p-value in the 1-lag autocorrelation test as shown in Table 2.2 and the best tail behaviors in the p-values comparison as shown in Figure 2.5a. It is not surprising that the PIs of the eOU-VIX model is least fluctuated, since the VIX is slighly less volatile compared to the Heston volatility as shown in Figure 2.1. For the 10-step prediction, the Heston model also appears to be the best judging by the diagnostic plots in Figure 2.5b and the percentage of the OV in Table 2.1. In general, the Heston model has the best performance judging by our goodness-of-fit metrics, relative to the other models considered in

this section.

2.3.3 Experiment for the SUR Model with Empirical Transformation

In this section, we apply the empirical transformation as described in Section 2.2.2 to the discretized eOU model (2.5), which we refer to as the E-eOU-VIX model, and compare it to the Heston model and the GARCH(1,1) model with Gaussian innovation. Compare to the Gen-VIX model, we do not need to worry about choosing $(l, \boldsymbol{\lambda})$.

Repeating the analysis of the previous Subsection, 95% prediction intervals and normalized Rosenblatt residuals are plotted for the three models in Figures 2.6 and 2.7, for 1-step and 10-step forecasts respectively. The 95% prediction intervals of the E-eOU-VIX model and the Heston model are very similar for both of the 1-step and 10-step forecasts, while the GARCH model predicts a little wider. In the plots for the normalized Rosenblatt residual, the curve of the E-eOU-VIX model is equivalent to that of the Heston model in the 10-step forecast and slightly better to than of the Heston model in the 1-step forecast. The GARCH model curves are worse than those of the other two model in both of the 1-step and 10-step forecasts. The numbers of the observations outside the supports of the empirical predictive distribution of the E-eOU-VIX model show significant improvement over those of the eOU-VIX model for both of the 1-step and 10-step forecasts, and those numbers are even smaller than the numbers of the Heston and the GARCH models given that the E-eOU-VIX's PIs are not significantly wider. This result means that the empirical transformation makes the model more robust to extreme observations.

The percentages of the OV for this study are reported in Table 2.3. The E-eOU-VIX model result is comparable to those of the Heston and the GARCH models in 1-step forecast and better in 10-step forecast. The result is mainly due to the empirical transformation, which effectively captures the skewness and the marginal heavy-tail features of the asset returns.

Table 2.4 reports the p-value of the Box-Ljung 1-lag autocorrelation test for 1-step forecast. The p-value for the E-eOU-VIX model is slightly smaller than those for the

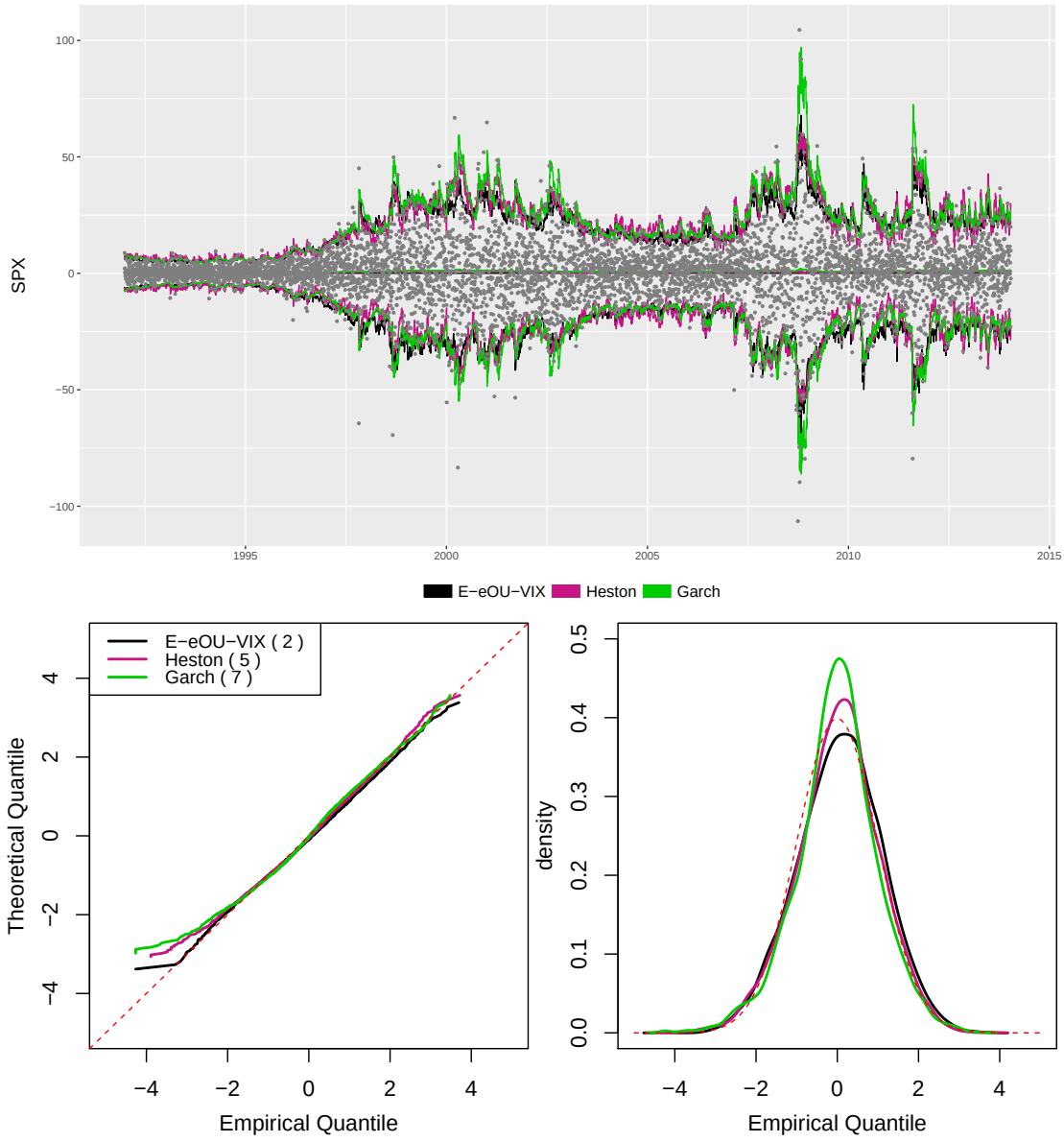


Figure 2.6: Diagnostic plots of 1-step prediction. *Top:* Grey dots are the true observations. All of the predictions and the observations are centered by the mean of the E-eOU-VIX prediction time point wise. *Bottom:* Red lines are the bench marks. The numbers in the legend shows the number of observations lie outside the supports of the empirical predictive distributions.

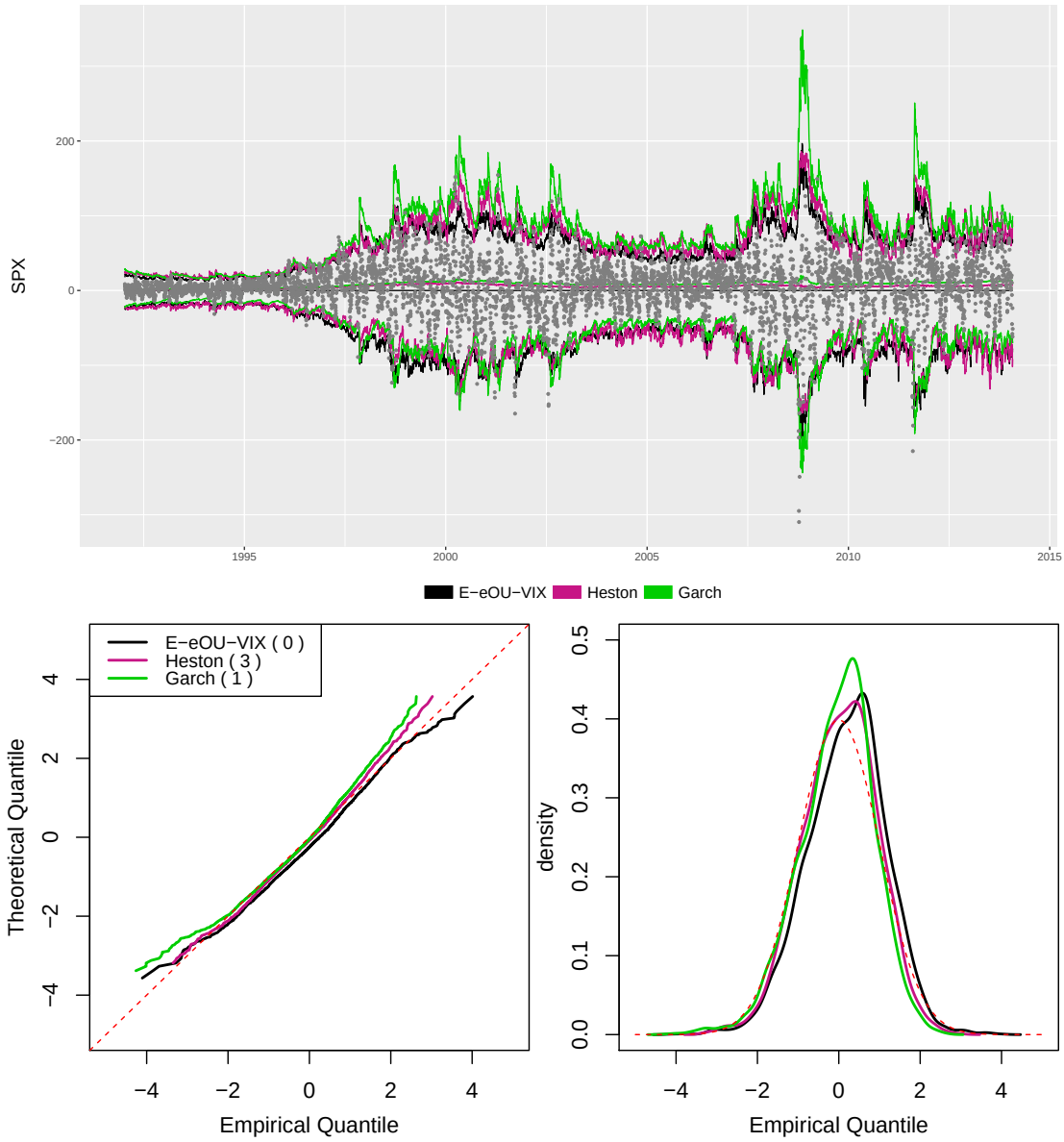


Figure 2.7: Diagnosis plots of 10-step prediction. *Top:* Grey dots are the true observations. All of the predictions and the observations are centered by the mean of the E-eOU-VIX prediction time point wise. *Bottom:* Red lines are the bench marks. The numbers in the legend shows the number of observations lie outside the supports of the empirical predictive distributions.

	E-eOU-VIX	Heston	GARCH
1-step Pred.	6.12 (0.32)	5.81 (0.32)	6.16 (0.32)
10-step Pred.	3.96 (0.26)	3.13 (0.24)	3.28 (0.24)

Table 2.3: Summary of the percentage of the OV for new model.

Heston and GARCH models.

	E-eOU-VIX	Heston model	GARCH(1,1)
1-step pred.	0.054	0.101	0.068

Table 2.4: Summary of the p-values from the 1-step Ljung-Box test for new model.

The empirical transformation is simple but very effective technique since the forecast result of the E-eOU-VIX model is much better than that of the eOU-VIX model. Compared to the Heston model, the E-eOU-VIX model makes relatively similar forecasts, which is more accurate than the GARCH model, especially for 10-step forecast. However, the main advantage of the E-eOU-VIX model is its low computation time. Table 2.5 reports the total computational time of the parameter estimation and 10-step prediction on the 5500 training datasets for each model considered in this experiment. The Gibbs sampler of the SUR model makes the inference of E-eOU-VIX model significantly faster than the MCMC algorithm of the Heston model and comparable to that of the GARCH model fitting. In summary, we use a simple and very fast model but obtain comparable forecast result to the Heston model.

	Heston model	E-eOU-VIX model	GARCH(1,1)
Volatility	latent	proxy & empirical trans.	conditional
Software	*msde* (C++)	custom implementation (C++)	*rugarch* (C++)
Running Time	$>$ 10 hours	$\approx$ 25 mins	$\approx$ 30 mins

Linux server with four Intel Xeon E5-4660v3 2.1 GHz 14-core CPUs

Table 2.5: Comparison of computational cost.

2.4 Conclusion

In this chapter, we have run an exploratory experiment within the SV framework using an observable volatility proxy. The out-of-sample predictions for more than 20 years of SPX data show that the E-eOU-VIX model not only is comparable to the Heston model but also can be analyzed much faster than the Heston model by the Gibbs sampler of the SUR model. In conclusion, this project mainly contributes three aspects. First, we explore the potential of the VIX as a proxy for the volatility of the SPX and find that it embeds important information. Second, we experiment on the SUR framework and find it is very flexible and easy to be analyzed. Third, we propose the empirical transformation, which leads to a very competitive modeling framework for the SPX.

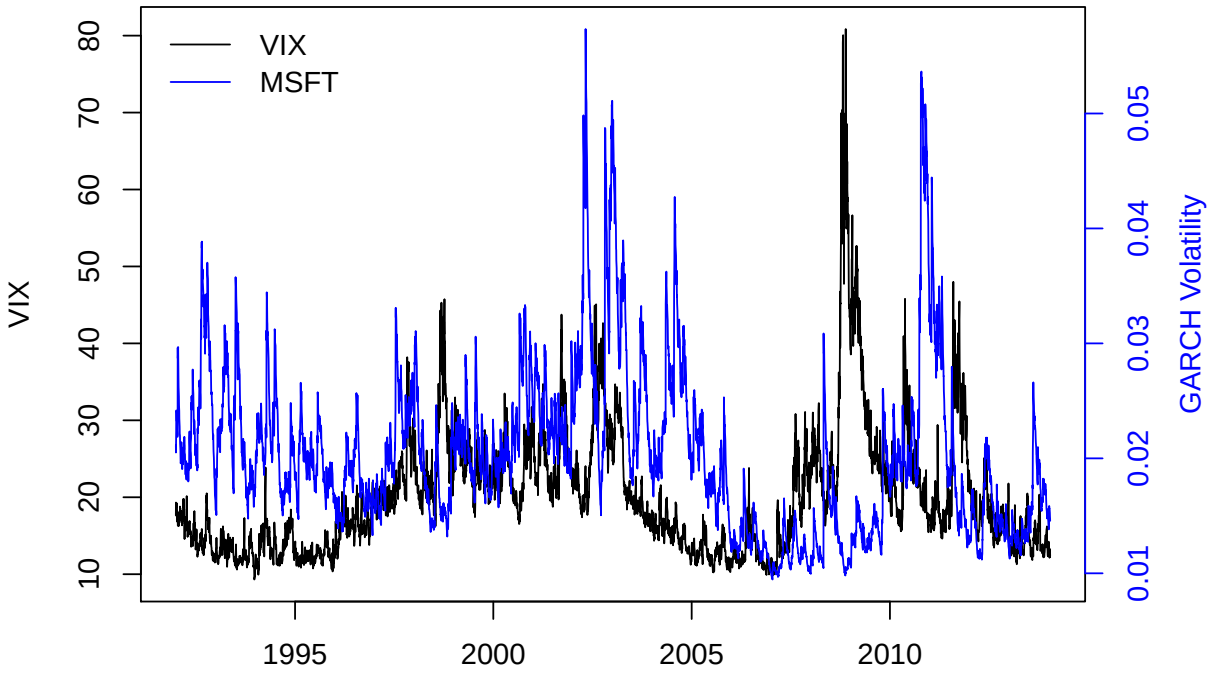


Figure 2.8: The VIX compared to the volatility estimators of the Microsoft stock by the standard GARCH(1,1) model.

The proposed framework works well for modeling the SPX using the VIX as the volatility proxy. While it is not expected to work as well for every individual stock since the VIX

and instantaneous volatility dynamics can differ considerably. For example, Figure 2.8 shows the GARCH(1,1) volatility of MSFT does not resemble the VIX. To solve this, we need more sophisticated methods, which is the topic of the upcoming chapter.

Chapter 3

Common-factor Stochastic Volatility Modeling with Observable Proxy

3.1 Introduction

Stochastic Volatility (SV) models have featured prominently in finance since the early work of Hull and White (1987); Wiggins (1987); Scott (1987). A generic SV model for an asset value S_t at time t is given by a two-dimensional stochastic differential equation (SDE) of the form

$$\begin{aligned} \mathrm{d}S_t &= \alpha S_t \,\mathrm{d}t + S_t V_t \,\mathrm{d}B_t^S \\ \mathrm{d}V_t &= \mu(V_t, \boldsymbol{\varphi}) \,\mathrm{d}t + \sigma(V_t, \boldsymbol{\varphi}) \,\mathrm{d}B_t^V, \end{aligned} \tag{3.1}$$

where the first line is seen to be the Black-Scholes model (Black and Scholes, 1973) but with time-dependent latent volatility V_t, which itself is a diffusion process correlated to S_t through the Brownian increments, such that $\mathrm{cor}(\mathrm{d}B_t^S, \mathrm{d}B_t^V) = \rho$. Several formulations for the volatility process have been proposed in the literature, e.g., Hull and White (1987); Heston (1993, 1997); Andersen and Lund (1997); Fouque et al. (2000).

SV models are especially useful for option pricing (e.g., Hull and White, 1987; Scott, 1987; Heston, 1993), as these derivatives often are mispriced under the constant-volatility

Black-Scholes model, especially deep in-the-money and out-of-the-money (Black, 1975). Compared to other non-constant volatility models such as the GARCH (Bollerslev, 1986), SV models are typically more flexible (Asai et al., 2006), capturing salient asset properties such as volatility clustering (Carr et al., 2003) and leverage effects (Bouchaud et al., 2001). A drawback of SV models is that generally their likelihood cannot be written in closed form; thus parameter inference and subsequent analyses often involve Markov chain Monte Carlo (MCMC) strategies (e.g., Broto and Ruiz, 2004; Golightly and Wilkinson, 2006; Andrieu et al., 2010; Kou et al., 2012; Beskos et al., 2013; Bladt et al., 2016; Kastner et al., 2017).

Since financial assets are known to be highly correlated with each other (e.g., Embrechts et al., 2002), multi-asset modeling is critical to applications such as portfolio optimization and risk management (Asai et al., 2006). A straightforward generalization of the single-asset model (3.1) to a multi-asset SV (mSV) model on q assets is

$$\begin{aligned} \mathrm{d}S_{it} &= \alpha_i S_{it}\,\mathrm{d}t + S_{it}V_{it}\,\mathrm{d}B_{it}^S \\ \mathrm{d}V_{it} &= \mu_i(V_{it}, \boldsymbol{\varphi}_i)\,\mathrm{d}t + \sigma_i(V_{it}, \boldsymbol{\varphi}_i)\,\mathrm{d}B_{it}^V, \qquad i = 1, \ldots, q, \end{aligned} \tag{3.2}$$

and we are left to specify a correlation structure between the $2 \times q$ Brownian innovations, namely

$$\boldsymbol{R} = \mathrm{cor}(\boldsymbol{B}_t^{SV}), \tag{3.3}$$

where $\boldsymbol{B}_t^{SV} = (\boldsymbol{B}_t^S, \boldsymbol{B}_t^V)$, $\boldsymbol{B}_t^S = (B_{1t}^S, \ldots, B_{qt}^S)$, and $\boldsymbol{B}_t^V = (B_{1t}^V, \ldots, B_{qt}^V)$. To this end, several possibilities have been explored in the literature, e.g., volatilities being independent of the assets (Harvey et al., 1994), independent of each other (Jacquier et al., 1995; Kastner et al., 2017), or conditionally independent given the assets (Szimayer et al., 2009). Many of these models have $\mathcal{O}(q^2)$ parameters, such that inference scales poorly to higher dimensions, compounding the challenge posed by multiple latent volatilities to efficient MCMC design. Perhaps more importantly, all the works above focus on modeling the dependence between assets, rather than their volatilities. However, there is considerable evidence in the financial literature of highly structured volatility dependence. For instance, Engle and Figlewski (2015) analyzed the implied volatilities of several components of the Standard and Poor's 500 (SPX) index, finding the VIX volatility index to be a significant common factor. Similarly, Herskovic et al. (2016) find a strong factor structure in the firm-level volatility of returns with yearly data since 1973.

In this chapter, we build upon such evidence of common-factor volatility structure to propose a hierarchical modeling framework for the specification of the correlation structure $\boldsymbol{R}$ in Equation (3.3). A fundamental feature of this approach is that the unobservable common volatility factor is readily proxied by observable measures of market volatility (such as the VIX). Our findings suggest this information greatly increases the precision of option price estimates, at little to no cost in forecasting accuracy, comparing to the model with unconstrained correlation structure.

The remainder of the chapter is organized as follows. In Section 3.2, we introduce the proposed mSV common-factor model and its volatility proxy. Section 3.3 presents a simulation study assessing the utility of the proxy. In section 3.4, we compare the proposed mSV model to several more complex dependence models for price-forecasting and option-pricing. Section 3.5 concludes the findings.

3.2 Correlation Model and Volatility Proxy

In the following section, we shall develop a correlation matrix $\boldsymbol{R} = \text{cor}(\boldsymbol{B}_t^{SV})$ (3.3) for the mSV model's Brownian innovations through a series of conditional distributions. Motivated by the findings of e.g., Engle and Figlewski (2015); Herskovic et al. (2016), we begin by formulating a common-factor model for the volatility innovations via

$$B_{it}^V = \tau_i B_{0t}^V + \sqrt{1-\tau_i^2}\, B_{it}^{\varepsilon}, \qquad i = 1, \ldots, q,$$

where B_{0t}^V is the common factor, B_{it}^{ε} are noise terms independent of B_{0t}^V and of each other, and $|\tau_i| < 1$. The construction is illustrated on the left-hand side of Figure 3.1 which indicates V_0 as the common volatility of $(V_1, \ldots, V_q)$.

To incorporate the leverage effect, each B_{it}^S is given marginal correlation ρ_i with its corresponding B_{it}^V, such that

$$B_{it}^S = \rho_i B_{it}^V + \sqrt{1-\rho_i^2}\, B_{it}^Z, \qquad i = 1, \ldots, q, \tag{3.4}$$

where $|\rho_i| < 1$ and B_{it}^Z are latent auxiliary Brownian motions independent of B_{it}^V.

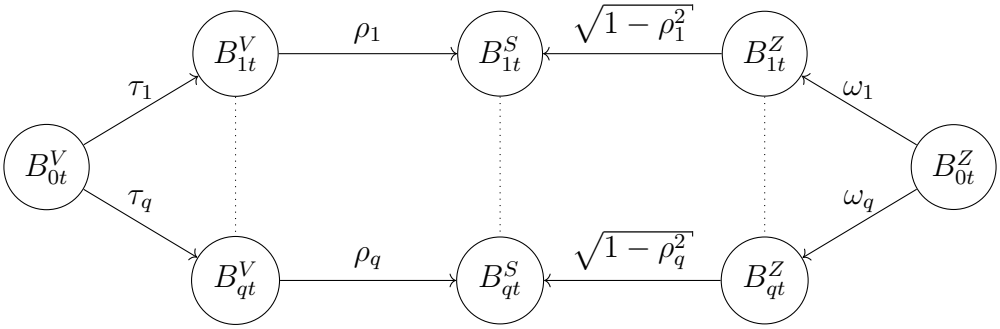


Figure 3.1: Graphical representation of the proposed correlation structure.

Since the q asset prices are observed, careful modeling of their correlation structure is required so as to be compatible with information provided by the data. Unlike many previous works wherein an unconstrained marginal or conditional correlation matrix is specified for $\boldsymbol{B}_t^S$, here we constrain it by assuming a factor model on $\boldsymbol{B}_t^Z = (B_{1t}^Z, \ldots, B_{qt}^Z)$ as shown on the right-hand side of Figure 3.1. Thus we can write B_{it}^Z as

$$B_{it}^Z = \omega_i B_{0t}^Z + \sqrt{1-\omega_i^2}\, B_{it}^\eta, \qquad i = 1, \ldots, q, \tag{3.5}$$

where $|\omega_i| < 1$, and B_{it}^η and B_{0t}^Z are Brownian motions all independent of each other. Substituting for B_{it}^Z in Equation (3.4) by Equation (3.5), we can express the asset innovations B_{it}^S as

$$B_{it}^S = \rho_i B_{it}^V + \sqrt{1-\rho_i^2}\left(\omega_i B_{0t}^Z + \sqrt{1-\omega_i^2}\, B_{it}^\eta\right), \qquad i = 1, \ldots, q.$$

By the hierarchical construction above, the correlation matrix $\boldsymbol{R} = \mathrm{cor}(\boldsymbol{B}_t^{SV})$ is guaranteed to be positive-definite, (see details in Appendix D) since it can be its entries are given by

$$\begin{aligned} &\mathrm{cor}(B_{it}^S, B_{it}^V) = \rho_i, \qquad \mathrm{cor}(B_{it}^V, B_{jt}^V) = \tau_i\tau_j, \qquad \mathrm{cor}(B_{it}^S, B_{jt}^V) = \rho_i\tau_i\tau_j, \\ &\mathrm{cor}(B_{it}^S, B_{jt}^S) = \rho_i\rho_j\tau_i\tau_j + \sqrt{(1-\rho_i^2)(1-\rho_j^2)}\,\omega_i\omega_j, \qquad 1 \le i \ne j \le q. \end{aligned} \tag{3.6}$$

Moreover, we highlight two additional properties of this correlation design that are of particular importance:

(i) As correlations between assets and volatilities are modeled entirely through the Brownian innovations $\boldsymbol{B}_t^{SV}$, the construction is compatible with any $\mu_i(V, \boldsymbol{\varphi}_i)$ and $\sigma_i(V, \boldsymbol{\varphi}_i)$

specifying the marginal volatilities of the mSV model (3.2).

(ii) While the correlation matrix $\boldsymbol{R}$ is not sparse, its design is parsimonious. That is, the total number of dependence parameters $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_q)$, $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_q)$ and $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_q)$ is $\mathcal{O}(q)$. Compared to $\mathcal{O}(q^2)$ for an unconstrained correlation matrix design, this is an attractive feature for scaling to higher dimensional settings.

3.2.1 Market Proxy for the Volatility Factor

In both single-asset and multi-asset SV modeling frameworks, the asset volatilities are latent states, which must be integrated out of the model to estimate parameters and option prices. However, many observable market quantities other than the asset prices themselves contain valuable volatility information for reducing estimation uncertainty. For single-asset SV models, the instantaneous volatility V_t is often "proxied" by the implied volatility of short-term maturities (Ledoit et al., 2002; Aït-Sahalia and Kimmel, 2007; Lewis, 2009), realized volatilities (McAleer and Medeiros, 2008), or estimators calculated from the high-frequency asset returns (e.g., Visser, 2008; Hansen and Lunde, 2011). A comparison of these and other proxies is found in Patton (2011).

In Chapter 2, the VIX is used as the proxy of the volatility of the SPX. In this chapter, we seek to proxy the common volatility factor of the mSV model with the VIX. The VIX is calculated by the Chicago Board Options Exchange (CBOE) according to the option prices of the S&P 500 (SPX) (Whaley, 1993). Its daily value can be interpreted as the market expectation of the SPX volatility for the next 30 days. The VIX is a widely-accepted measure of the near-future market risk (Whaley, 2009).

Engle and Figlewski (2015) report strong correlations between VIX and the implied volatilities of 28 SPX components. Further evidence in support of VIX as a good proxy for the SPX common volatility factor is presented in Figure 3.2. This Figure compares VIX to the GARCH(1,1) volatilities of the SPX index and five of its major components during the period of 2010-2015. After rescaling, we observe that the GARCH volatilities and the VIX proxy have similar dynamics.

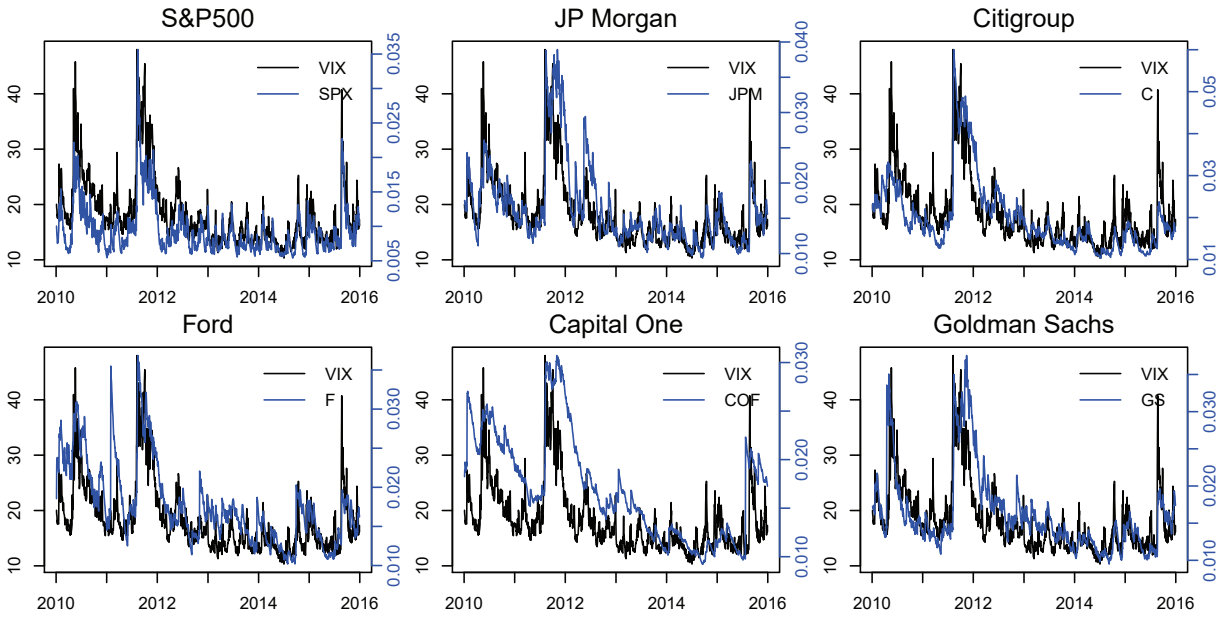


Figure 3.2: Comparison between VIX and GARCH volatilities for SPX and five of its major constituents.

To include information from the volatility proxy in the mSV model (3.2), we simply add an SDE process V_{0t} linking to the common volatility factor:

$$\begin{aligned} \mathrm{d}S_{it} &= \alpha_i S_{it}\,\mathrm{d}t + S_{it}V_{it}\,\mathrm{d}B^S_{it}, \\ \mathrm{d}V_{it} &= \mu_i(V_{it},\boldsymbol{\varphi}_i)\,\mathrm{d}t + \sigma_i(V_{it},\boldsymbol{\varphi}_i)\,\mathrm{d}B^V_{it}, \qquad i = 1,\ldots,q, \\ \mathrm{d}V_{0t} &= \mu_0(V_t,\boldsymbol{\varphi}_0)\,\mathrm{d}t + \sigma_0(V_{0t},\boldsymbol{\varphi}_0)\,\mathrm{d}B^V_{0t}, \end{aligned} \tag{3.7}$$

where the correlation matrix on $\boldsymbol{B}^\star_t = (\boldsymbol{B}^{SV}_t, B^V_{0t})$ is given by

$$\boldsymbol{R}^\star = \mathrm{cor}(\boldsymbol{B}^\star_t) = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{c}' \\ \boldsymbol{c} & 1 \end{bmatrix}, \tag{3.8}$$

where $\boldsymbol{R} = \mathrm{cor}(\boldsymbol{B}^{SV}_t)$ as in Equation (3.6) and $\boldsymbol{c} = (\rho_1\tau_1,\ldots,\rho_q\tau_q,\tau_1,\ldots,\tau_q)$. It is worth emphasizing that Equation (3.7) gives considerable flexibility to choose $\mu_0(V,\boldsymbol{\varphi}_0)$ and $\sigma_0(V,\boldsymbol{\varphi}_0)$ to best fit the observed proxy V_{0t}. We point out that, unlike other proxy-based approaches (e.g., Aït-Sahalia and Kimmel, 2007; Visser, 2008), our volatility proxy is NOT

of plug-in type. That is, the proxy need only provide information about the volatility innovations, rather than the volatilities themselves. Thus, the choice of proxy in this setting is much less important than in the plug-in context.

For this design of the correlation matrix (3.8), we need to mention several special cases concerning the identifiability of ω_i. For $q = 1$, there is no need of the parameter ω_1. For $q = 2$, we cannot distinguish ω_1 from ω_2, instead we only need to identify $\omega_1\omega_2$. For $q \geq 3$, no identifiable issue exists.

3.3 Simulation Study

In this section, we wish to quantify the extra information provided by the volatility proxy when the common-factor mSV model (3.7-3.8) is correct. To this end, we simulate data from a multivariate exponential Ornstein-Uhlenbeck (eOU) model having the form of

$$\begin{aligned} \mathrm{d}S_{it} &= \alpha_i S_{it}\,\mathrm{d}t + S_{it}V_{it}\,\mathrm{d}B^S_{it}, \\ \mathrm{d}\log V_{it} &= \gamma_i(\mu_i - \log V_{it})\,\mathrm{d}t + \sigma_i\,\mathrm{d}B^V_{it}, \qquad i = 1,\ldots,q, \\ \mathrm{d}\log V_{0t} &= \gamma_0(\mu_0 - \log V_{0t})\,\mathrm{d}t + \sigma_0\,\mathrm{d}B^V_{0t}, \end{aligned} \tag{3.9}$$

with common-factor correlation structure given by $\boldsymbol{R}^\star$ in Equation (3.8). The single-asset eOU model was introduced by Scott (1987); Fouque et al. (2000), along with its mathematical properties and option-pricing scheme. A more common choice for each marginal SV model is perhaps that of Heston (1993), for which the volatility diffusions are given by $\mathrm{d}V^2_{it} = \gamma_i(\mu_i - V^2_{it})\,\mathrm{d}t + \sigma_i V_{it}\,\mathrm{d}B^V_{it}$. However, empirical evidence indicates that the two models are similar (e.g. Perelló et al., 2004; Cisana et al., 2007), with the eOU model being somewhat more numerically tractable as explained in the end of Appendix B.

We generate $N = 1500$ observations of the eOU-mSV model (3.9), with $q = 4$ assets and interobservation time of $\Delta\mathrm{T} = 1/252$ years (there are 252 trading days in a year). The true parameters of the model are given in Table 3.1. Inference is then conducted conditional on two settings of observable data:

1. With *observed volatility factor* (OVF), where the asset values $\boldsymbol{S} = \{S_{it} : i = 1,\ldots,q,\ t = \Delta\mathrm{T},\ldots,N\Delta\mathrm{T}\}$ and the common-factor volatilities $\boldsymbol{V}_0 = (V_{0,\Delta\mathrm{T}},\ldots,V_{0,N\Delta\mathrm{T}})$

α_1	α_2	α_3	α_4	γ_1	γ_2	γ_3	γ_4	μ_1	μ_2	μ_3	μ_4
0.1	0.04	0.09	0.03	4	3	2.5	8	-1.5	-1.3	-1.5	-1.7
σ_1	σ_2	σ_3	σ_4	ρ_1	ρ_2	ρ_3	ρ_4	τ_1	τ_2	τ_3	τ_4
1.2	1	0.75	2	-0.68	-0.7	-0.65	-0.68	0.85	0.9	0.93	0.7
ω_1	ω_2	ω_3	ω_4	γ_0	μ_0	σ_0					
0.94	0.9	0.8	0.65	6.5	2.8	1.2					

Table 3.1: The true parameters for the simulation study.

are observed and the individual volatilities $\boldsymbol{V} = \{V_{it} : i = 1, \ldots, q,\ t = \Delta\mathrm{T}, \ldots, N\Delta\mathrm{T}\}$ are latent.

2. With *unobserved volatility factor* (UVF), where only the asset values $\boldsymbol{S}$ is observed, and both the common volatility factor $\boldsymbol{V}_0$ and the individual volatilities $\boldsymbol{V}$ are latent.

3.3.1 Bayesian Inference

Focusing on the OVF setting, the likelihood for the model parameters $\boldsymbol{\theta} = (\boldsymbol{\alpha}, \boldsymbol{\mu}, \boldsymbol{\gamma}, \boldsymbol{\sigma}, \boldsymbol{\tau}, \boldsymbol{\rho}, \boldsymbol{\omega})$ with $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_q)$, $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_q)$, etc., is given by

$$\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{S}, \boldsymbol{V}_0) = \int \prod_{n=1}^{N-1} p(\boldsymbol{Y}_{(n+1)\Delta\mathrm{T}}|\boldsymbol{Y}_{n\Delta\mathrm{T}}, \boldsymbol{\theta})\, \mathrm{d}\boldsymbol{V},$$

where $\boldsymbol{Y}_{n\Delta\mathrm{T}} = (\boldsymbol{S}_{n\Delta\mathrm{T}}, \boldsymbol{V}_{n\Delta\mathrm{T}}, V_{0,n\Delta\mathrm{T}})$. This likelihood function presents two challenges for parameter inference; namely, that the SDE transition density $p(\boldsymbol{Y}_{t+\Delta\mathrm{T}}|\boldsymbol{Y}_t, \boldsymbol{\theta})$ is not available in closed form, and that the latent volatilities $\boldsymbol{V}$ cannot be analytically integrated out. The first challenge is typically resolved employing an Euler-Maruyama approximation (Maruyama, 1955). That is, for a general multivariate diffusion process $\mathrm{d}\boldsymbol{Y}_t = \boldsymbol{\Lambda}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t)\,\mathrm{d}t + \boldsymbol{\Psi}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t)\,\mathrm{d}\boldsymbol{B}_t$, for small interobservation time $\Delta\mathrm{T}$ the transition density is approximated as

$$\boldsymbol{Y}_{t+\Delta\mathrm{T}}|\boldsymbol{Y}_t \approx \mathcal{N}\big(\boldsymbol{Y}_t + \boldsymbol{\Lambda}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t)\Delta\mathrm{T}, \boldsymbol{\Psi}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t)\boldsymbol{\Psi}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t)'\Delta\mathrm{T}\big), \tag{3.10}$$

such that the approximate likelihood is given by

$$\hat{\mathcal{L}}(\theta|\boldsymbol{S}, \boldsymbol{V}_0) = \int \prod_{n=1}^{N-1} f(\boldsymbol{Y}_{(n+1)\Delta\mathrm{T}}|\boldsymbol{Y}_{n\Delta\mathrm{T}}, \boldsymbol{\theta})\, \mathrm{d}\boldsymbol{V},$$

where $f(\boldsymbol{Y}_{(n+1)\Delta\mathrm{T}}|\boldsymbol{Y}_{n\Delta\mathrm{T}}, \boldsymbol{\theta})$ is the PDF of the multivariate normal specified by Equation (3.10).

The second challenge of integrating out the latent volatilities is often accomplished in a Bayesian data augmentation setting. That is, upon selecting a prior $\pi(\boldsymbol{\theta})$, an MCMC algorithm is run on the augmented (approximate) posterior distribution

$$p(\boldsymbol{\theta}, \boldsymbol{V}|\boldsymbol{S}, \boldsymbol{V}_0) \propto \pi(\boldsymbol{\theta}) \cdot \prod_{n=1}^{N-1} f(\boldsymbol{Y}_{(n+1)\Delta\mathrm{T}}|\boldsymbol{Y}_{n\Delta\mathrm{T}}, \boldsymbol{\theta}), \tag{3.11}$$

whereupon marginalizing the MCMC output over either parameters or volatilities produces draws from either $p(\boldsymbol{V}|\boldsymbol{S}, \boldsymbol{V}_0)$ or $p(\boldsymbol{\theta}|\boldsymbol{S}, \boldsymbol{V}_0)$.

To sample from the augmented posterior distribution (3.11), we use the MCMC algorithm implemented in the R package *msde* (Lysy and Tong, 2017), of which the details are provided in Appendix B. The prior we have used is

$$\begin{aligned} &\pi(\boldsymbol{\alpha}, \boldsymbol{\mu}, \log \boldsymbol{\sigma}, \log \boldsymbol{\gamma}, \log \tfrac{\boldsymbol{\rho}+1}{1-\boldsymbol{\rho}}, \log \tfrac{\boldsymbol{\tau}+1}{1-\boldsymbol{\tau}}, \log \tfrac{\boldsymbol{\omega}+1}{1-\boldsymbol{\omega}}) \propto 1 \\ \iff \quad &\pi(\boldsymbol{\theta}) \propto \prod_{i=1}^{q} \left[\gamma_i \sigma_i (1-\rho_i^2)(1-\tau_i^2)(1-\omega_i^2)\right]^{-1}. \end{aligned}$$

Because the MCMC output is highly correlated, 1,000,000 posterior iterations are recorded after 10,000 iterations of burn-in.

For the UVF setting, the likelihood function is

$$\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{S}) = \int \prod_{n=1}^{N-1} p(\boldsymbol{Y}_{(n+1)\Delta\mathrm{T}}|\boldsymbol{Y}_{n\Delta\mathrm{T}}, \boldsymbol{\theta})\, \mathrm{d}\boldsymbol{V}\, \mathrm{d}\boldsymbol{V}_0, \tag{3.12}$$

While at first glance this increases the number of latent variables, in fact Equation (3.7) specifies a marginal SDE for $(\boldsymbol{S}_t, \boldsymbol{V}_t)$. Thus, $\boldsymbol{V}_0$ in Equation (3.12) is analytically integrated out, and Bayesian inference for the UVF model becomes nearly identical to the OVF case.

3.3.2 Posterior Distribution of Latent Volatilities

Figure 3.3 displays the posterior distributions $p(V_{i,n\Delta T}|\boldsymbol{S})$ and $p(V_{i,n\Delta T}|\boldsymbol{S}, \boldsymbol{V}_0)$ corresponding to the UVF and OVF settings, for a random selection of observations in $n = 1, \ldots, 1500$. The vertical dotted lines represent the true volatility values generated by the data sim-

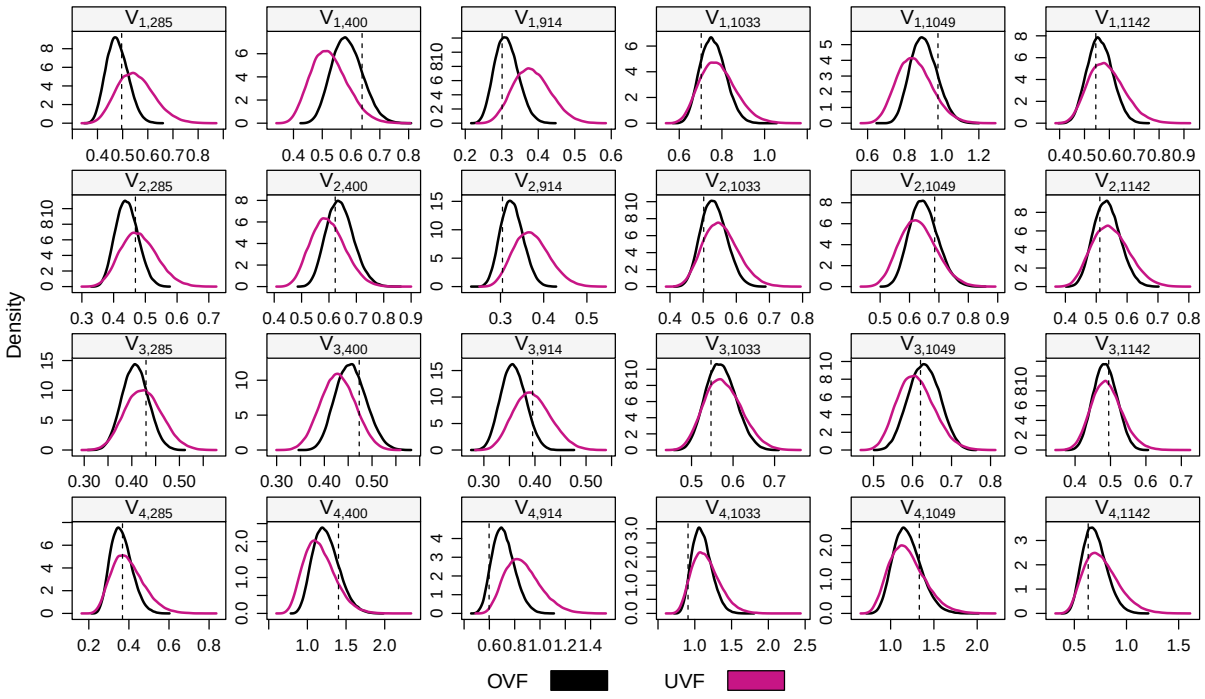


Figure 3.3: The posterior densities of the volatilities on some randomly chosen time points. Dashed lines mark the true simulated values.

ulation. In most cases in Figure 3.3, the OVF mode is closer than that of UVF to the true volatility value, and in all cases the OVF standard deviation is smaller. Quantitative summaries of these findings are reported in Table 3.2. Each entry of the table is the ratio of the mean relative errors, which is calculated as

$$\sum_{n=1}^{N}(e_{in}^{\text{UVF}}/V_{i,n\Delta T}^{\text{true}})/\sum_{n=1}^{N}(e_{in}^{\text{OVF}}/V_{i,n\Delta T}^{\text{true}}),$$

UVF/OVF	$\sqrt{\text{Bias}}$	s.d.	RMSE
V_1	1.17	1.38	1.29
V_2	1.25	1.39	1.33
V_3	1.49	1.31	1.38
V_4	1.15	1.31	1.24

Table 3.2: UVF to OVF ratio of various statistics of the posterior volatilities.

where $V^{\text{true}}_{i,n\Delta\text{T}}$ is the true value of the volatility produced by the simulation, e^{OVF}_{in} is alternatively the square root of the bias,

$$e^{\text{OVF}}_{in} = \left| E[V_{i,n\Delta\text{T}} | \boldsymbol{S}, \boldsymbol{V}_0] - V^{\text{true}}_{i,n\Delta\text{T}} \right|,$$

the posterior standard deviation,

$$e^{\text{OVF}}_{in} = \text{sd}(V_{i,n\Delta\text{T}} | \boldsymbol{S}, \boldsymbol{V}_0),$$

and the root mean square error (RMSE),

$$e^{\text{OVF}}_{in} = \sqrt{E[(V_{i,n\Delta\text{T}} - V^{\text{true}}_{i,n\Delta\text{T}})^2 | \boldsymbol{S}, \boldsymbol{V}_0]},$$

and we have analogous definitions for e^{UVF}_{in} but conditioning only on $\boldsymbol{S}$. The expectation represents the evaluation of the Monte Carlo average over the posterior sample. Table 3.2 indicates that OVF can reduce the average value of these statistics by 20-30%.

3.3.3 Posterior Distribution of Parameters

Figure 3.4 compares the OVF and UVF posteriors for the volatility parameters $\boldsymbol{\gamma}$, $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, $\boldsymbol{\rho}$, $\boldsymbol{\tau}$, and $\boldsymbol{\omega}$, with true parameter values indicated by the vertical dotted lines. The extra information provided by OVF is especially apparent for the correlation parameters $\boldsymbol{\rho}$, $\boldsymbol{\tau}$, and $\boldsymbol{\omega}$. Numerically this is confirmed by the ratio of parameter-wise RMSEs displayed in Table 3.3, which typically indicates a 1.5-5x decrease in RMSE by conditioning on the volatility proxy.

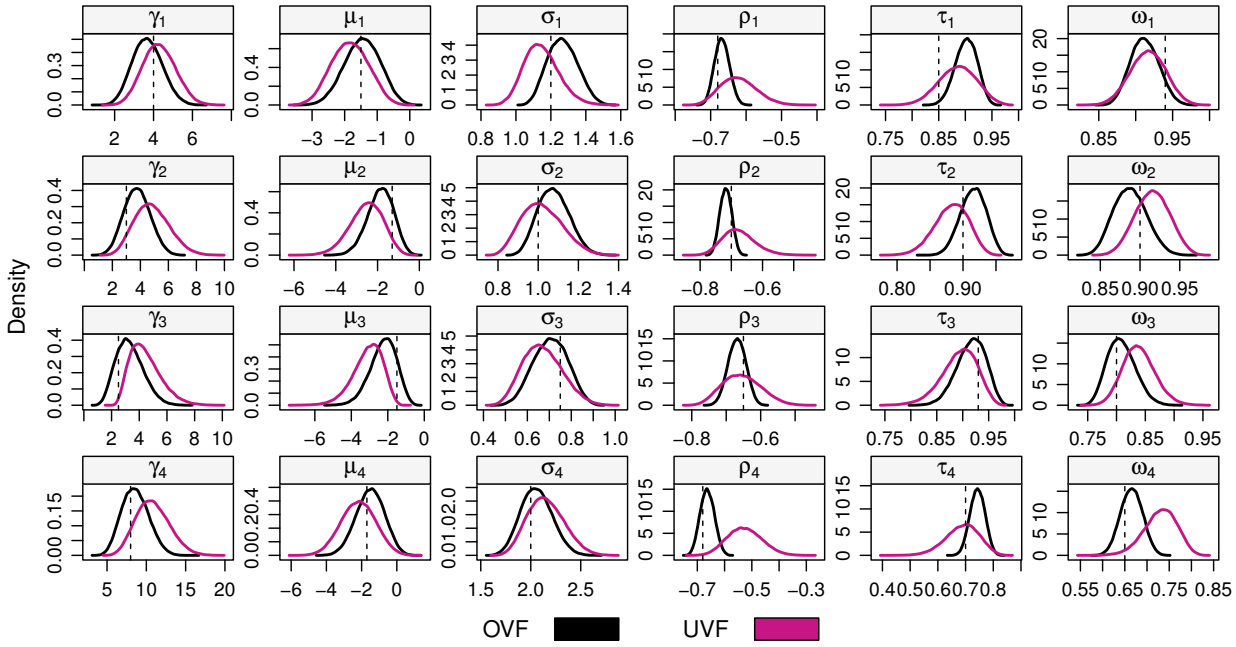


Figure 3.4: The posterior densities of some selected parameters.

UVF/OVF	α	γ	μ	σ	ρ	τ	ω
Asset 1	1.48	1.16	1.42	1.10	10.18	0.83	1.00
Asset 2	1.29	3.31	3.08	0.88	5.16	1.62	1.19
Asset 3	1.23	3.01	2.97	1.79	3.08	2.67	3.36
Asset 4	1.89	3.76	1.74	1.98	28.26	1.54	9.01

Table 3.3: The ratio of the RMSE of the posterior parameters.

3.4 Applications to Portfolio Management

Consider a stock portfolio to be composed of $q = 4$ prominent constituents of SPX in the financial sector: JP Morgan (JPM), Citigroup (C), Goldman Sachs (GS) and Capital One Financial (COF). Our period of analysis is $N = 2500$ daily observations (i.e., $\Delta T = 1/252$ years) between October 2007 and September 2017. In the following analyses we compare various eOU-mSV models of the form (3.9) specified by three features:

1. *Correlation structure:* either the common-factor (FAC) model (3.8), or a fully unconstrained (UNC) correlation matrix on $\boldsymbol{B}_t^\star = (\boldsymbol{B}_t^{SV}, B_{0t}^V)$.

2. *Volatility Factor:* whether it is latent (LAT) or proxied (PROX) by the VIX.

3. *Number of assets:* depending on the quantity of interest, q is either 1 or 4.

As in Section 3.3, we use the R package *msde* (Lysy and Tong, 2017) to conduct Bayesian inference. For the UNC model, a flat prior is assumed for the correlation matrix of the Brownian innovations ($\boldsymbol{R}$ or $\boldsymbol{R}^\star$).

3.4.1 Forecasting Analysis

Let $\boldsymbol{S}_n = (S_{1n}, \ldots, S_{qn})$ denote the value of each asset on day n, and define similar notations for $\boldsymbol{V}_n$ and $\boldsymbol{V}_{0n}$. Suppose that a portfolio on these q assets is given by

$$\mathcal{P}_n = \sum_{i=1}^{q} w_{in} S_{in}, \tag{3.13}$$

where we have allowed the weight composition $\boldsymbol{w}_n = (w_{1n}, \ldots, w_{qn})$ of the portfolio to change from day to day. Then the value of portfolio (3.13) on day $n+1$ is

$$\mathcal{P}_{n+1} = \sum_{i=1}^{q} w_{in} S_{i,n+1}.$$

Thus, given a portfolio balancing strategy which determines the weight composition $\boldsymbol{w}_n$ on day n, a Bayesian rolling-window forecast of $\mathcal{P}_{n+1}$ is conducted in the following algorithm 3.1. In the following subsections, we compare the forecasts of several mSV models for two portfolios of interest. In both cases, the training window is of $W = 1500$ days, such that $N_{\text{test}} = 1000$ testing days are used to evaluate the quality of the forecasts.

Single-Asset Portfolio

In this first experiment we consider a portfolio consisting entirely of JPM. Four eOU-mSV models are compared:

Algorithm 3.1: Bayesian Rolling-window Forecasting

1 Suppose we employ a window of W days to train an mSV model. Then on a given day n, the training data consists of

$$\boldsymbol{Y}_n^W = (\boldsymbol{Y}_{n-W+1}, \ldots, \boldsymbol{Y}_n), \tag{3.14}$$

where $\boldsymbol{Y}_n = (\boldsymbol{S}_n, V_{0,n})$ when the common-factor volatility is proxied, and $\boldsymbol{Y}_n = \boldsymbol{S}_n$ when it is latent.

2 Use MCMC to sample M draws from the posterior distribution $p(\boldsymbol{V}_n, \boldsymbol{\theta}|\boldsymbol{Y}_n^W)$.

3 For each draw $(\boldsymbol{V}_n^{(m)}, \boldsymbol{\theta}^{(m)})$ obtained in Step 2, produce the corresponding draw $\boldsymbol{S}_{n+1}^{(m)} \sim p(\boldsymbol{S}_{n+1}|\boldsymbol{V}_n^{(m)}, \boldsymbol{\theta}^{(m)}, \boldsymbol{Y}_n)$.

4 Setting $\mathcal{P}_{n+1}^{(m)} = \sum_{i=1}^{q} w_{in} S_{i,n+1}^{(m)}$, the collection of all M of these values is a sample from the Bayesian forecast distribution

$$p(\mathcal{P}_{n+1}|\boldsymbol{Y}_n^W) = \int p(\mathcal{P}_{n+1}|\boldsymbol{V}_n, \boldsymbol{\theta}, \boldsymbol{Y}_n) \cdot p(\boldsymbol{V}_n, \boldsymbol{\theta}|\boldsymbol{Y}_n^W)\, \mathrm{d}\boldsymbol{V}_n\, \mathrm{d}\boldsymbol{\theta}.$$

5 Replace n by $n+1$ and repeat Steps 1-4 for N_{test} days. On each of these testing days, the forecast distribution of $\mathcal{P}_{n+1}$ can be compared to the realized value of the portfolio, $\mathcal{P}_{n+1}^{\text{obs}}$.

1. The original single-asset eOU model of Scott (1987); Fouque et al. (2000). In our notation, this corresponds to a 1-LAT-UNC model.

2. The single-asset UNC model with proxy (1-PROX-UNC). That is, we borrow information from the VIX innovations to estimate those of the latent JPM volatility, but without imposing the factor model structure.

3. The single-asset FAC model with proxy (1-PROX-FAC), to assess the impact of the factor correlation structure.

4. The 4-asset FAC model with proxy (4-PROX-FAC), to assess the impact of multivariate modeling for single-asset forecasts.

Figure 3.5 compares the distributions of JPM daily return forecasts,

$$\mathcal{R}_{n+1} = \frac{S_{\text{JPM},n+1} - S_{\text{JPM},n}}{S_{\text{JPM},n}},$$

among the four mSV models. Also compared are the posterior distributions of the latent volatilities $p(V_{\text{JPM},n}|\boldsymbol{Y}_n^W)$. The volatility proxy appears to have a significant impact on

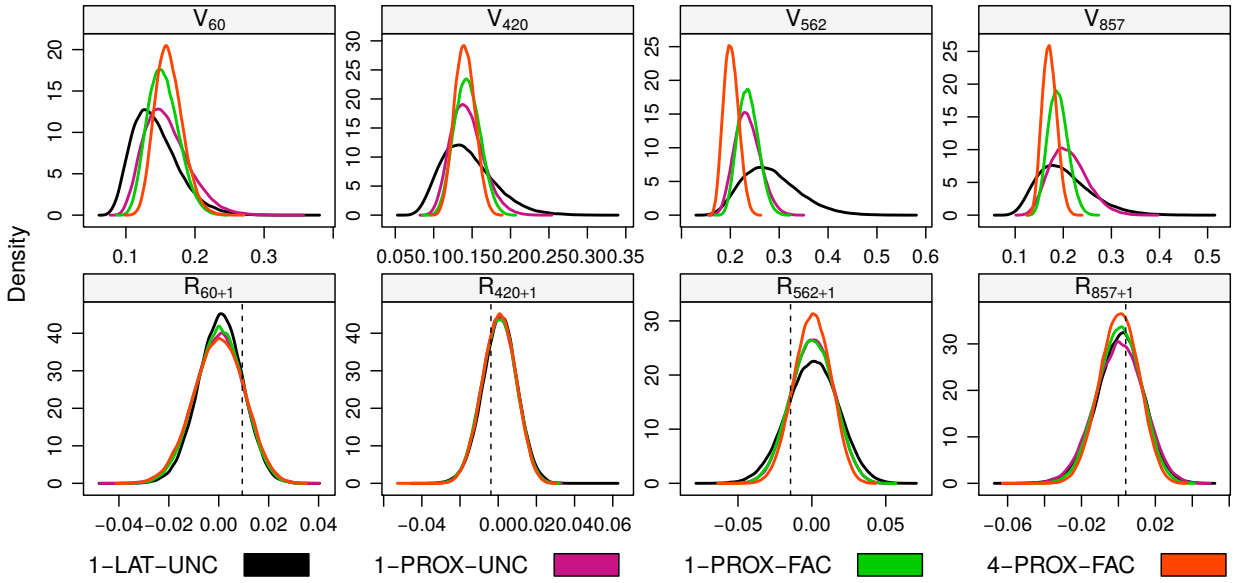


Figure 3.5: Posterior distributions of JPM return forecasts and latent volatilities for selected testing days. True return values are indicated by the dotted lines.

the precision of the latent volatility estimates, as does multivariate modeling within the common-factor (FAC) framework. However, the differences between models are much less significant when it comes to forecasting the JPM returns.

A different view of these results is presented in Figure 3.6. The top half of the Figure displays the mean and 95% prediction interval of the return forecasts for each model. While the prediction means are essentially identical, the prediction intervals are considerably

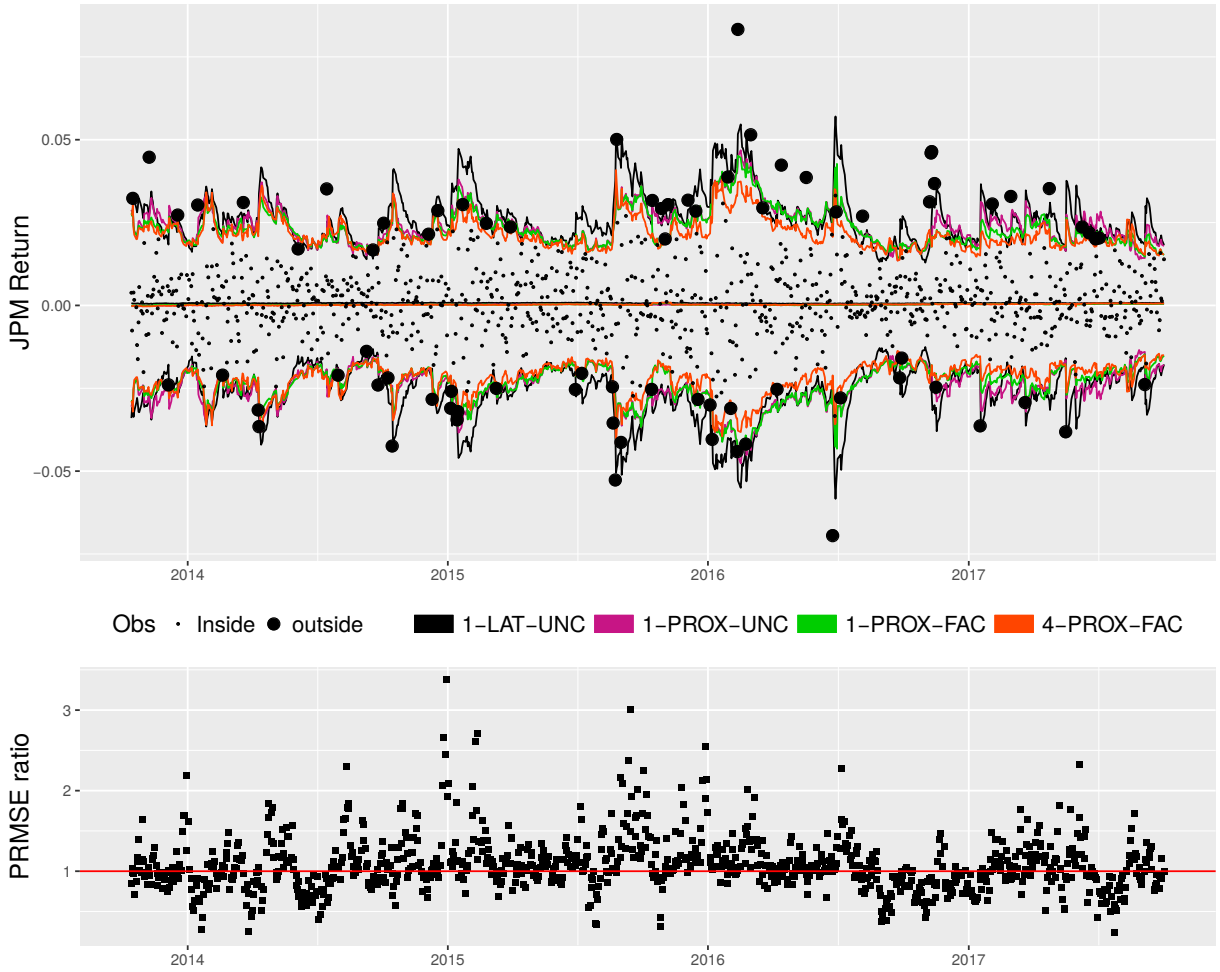


Figure 3.6: *Top:* Posterior means and 95% prediction intervals for the JPM return forecasts of each model. True return values are denoted by circles. *Bottom:* Ratio of 1-LAT-UNC model PRMSEs to those of 4-PROX-FAC.

narrower for the 4-PROX-FAC model during periods of higher JPM volatility. Seemingly, this is because individual stocks are more volatile than the market average, such that extreme individual returns have less power to increase forecast intervals when borrowing information from an overall market volatility index such as the VIX. To quantify this effect,

the bottom of Figure 3.6 displays the ratio of predictive root mean square error,

$$\text{PRMSE} = \sqrt{E[(\mathcal{R}_{n+1} - \mathcal{R}_{n+1}^{\text{obs}})^2 | \boldsymbol{Y}_n^W]},$$

between the 1-LAT-UNC and 4-PROX-FAC models. About 70% of these ratios are between 1-4, indicating that the 4-PROX-FAC forecast distributions generally are more concentrated around the true return value than those of 1-LAT-UNC.

Table 3.4 compares the true coverage of various α-level prediction intervals to their nominal value. That is, for a given level α, the Table reports the fraction of $N_{\text{test}} = 1000$ testing days for which the true return was inside the prediction interval. Overall, the

Nominal Percentage	10	20	30	40	50	60	70	80	90	95	99
1-LAT-UNC	13	24	36	48	58	66	74	84	91	95	98
1-PROX-UNC	12	25	37	48	57	65	73	82	90	94	98
1-PROX-FAC	12	25	36	48	57	66	74	81	90	94	97
4-PROX-FAC	11	23	35	45	54	63	71	78	88	93	96

Table 3.4: True coverage of Bayesian prediction intervals for JPM returns. All of the numbers are in percentage scale.

single-asset models tend to overcover at nominal levels of $\alpha \leq 70\%$, whereas the 4-PROX-FAC model exhibits slightly more undercoverage at nominal levels of $\alpha \geq 80\%$. That being said, the true coverage levels of all models are typically within 5% of each other and of the nominal α, indicating reasonably similar and accurate model forecasts. Even for single-asset forecasts, borrowing market information through the common-factor model and VIX proxy improves the PRMSE, with little price to pay in terms of coverage probability.

Equally-Weighted Portfolio

In this next experiment, we consider a so-called "equally-weighted" portfolio which is constructed as follows. Suppose that on day n the portfolio has a value of C. Then the portfolio is rebalanced such that C/q dollars are invested in each asset. In other words,

the weight asset i is $w_{in} = C/(qS_{in})$, and the relative return of the portfolio on day $n+1$ is given by

$$\mathcal{R}_{n+1} = \frac{1}{q}\sum_{i=1}^{q}\frac{S_{i,n+1} - S_{i,n}}{S_{i,n}}.$$

The analysis of Section 5 is repeated here, comparing three different models:

1. The 4-asset UNC model with no volatility proxy (LAT).
2. The 4-asset UNC model with volatility proxy (PROX).
3. The 4-asset FAC model with no volatility proxy (LAT).
4. The 4-asset FAC model with volatility proxy (PROX).

In this case, the forecast distributions of 4-LAT-UNC, 4-PROX-UNC and 4-LAT-FAC are very close to each other for most of the time points as we can see in Figure 3.8, indicating that the unconstrained correlation design does not channel much information from the proxy into return predictions and the factor correlation design does not make much difference without the proxy information. The green lines of the 4-LAT-FAC almost overlay on the lines of the first two models. On the other hand, the prediction intervals of the 4-PROX-FAC model are considerably narrower than those of the unconstrained models, especially during the period after 2017. Consequently, the PRMSEs of the common-factor model is considerably lower than those of the unconstrained model alternatives, and conversely, exhibits somewhat more undercoverage of its prediction intervals at nominal $\alpha \geq 80\%$ than we observed in the single-asset forecasting experiment as we can see in Table 3.5.

3.4.2 Option Pricing Experiment

An option contract on a specific asset gives its owner the right – but not the obligation – to buy or sell the asset at an agreed-upon price and future date. These financial derivatives are an indispensable tool for portfolio managers to mitigate the risk of holding volatile assets, by offsetting a potential loss by purchasing the option to sell the asset at a higher price.

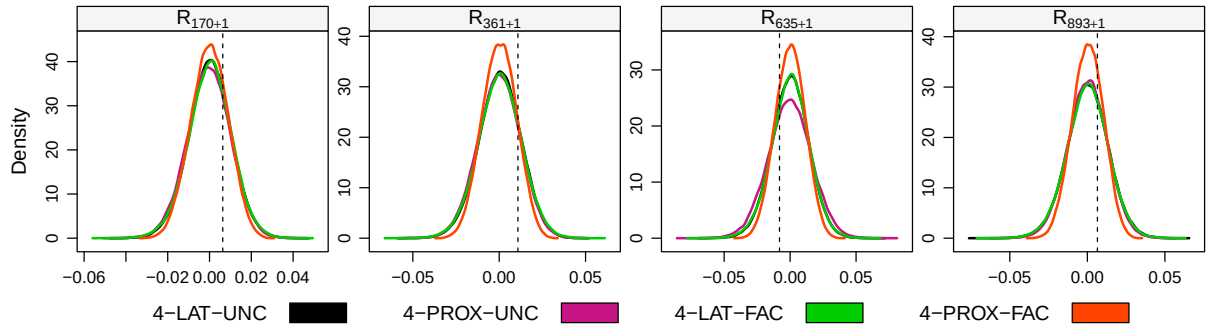


Figure 3.7: Posterior distributions of the equally-weighted portfolio return forecasts for selected testing days. True return values are indicated by the dotted lines.

Nominal	10	20	30	40	50	60	70	80	90	95	99
4-LAT-UNC	12	24	33	43	53	63	71	80	89	94	99
4-PROX-UNC	13	23	34	43	53	62	71	80	89	94	99
4-LAT-FAC	13	25	33	44	53	64	72	81	90	94	99
4-PROX-FAC	12	24	32	42	52	61	69	77	86	91	97

Table 3.5: True coverage of Bayesian prediction intervals for equally-weighted portfolio returns. All of the numbers are in percentage scale.

Correctly pricing option contracts is perhaps the most important application of SDEs in finance. For an asset S_t which follows the dynamics of a univariate SV model (3.1), the option price O_t is given by

$$O_t = \mathcal{G}(S_t, V_t, \boldsymbol{\theta}, r, K, T),$$

where V_t is the assets instantaneous volatility, $\boldsymbol{\theta}$ are the SV model parameters, r is the risk-free rate, and K and $T - t$ are the strike price and time-to-maturity specified by the contract. The function $\mathcal{G}$ depends on the type of option contract. For example, American-style options allow the holder to buy or sell the asset at strike price K anytime before maturity. In this section, we shall examine the ability of various mSV models to price various American option contracts on the JPM stock on the $N_{\text{test}} = 500$ days between

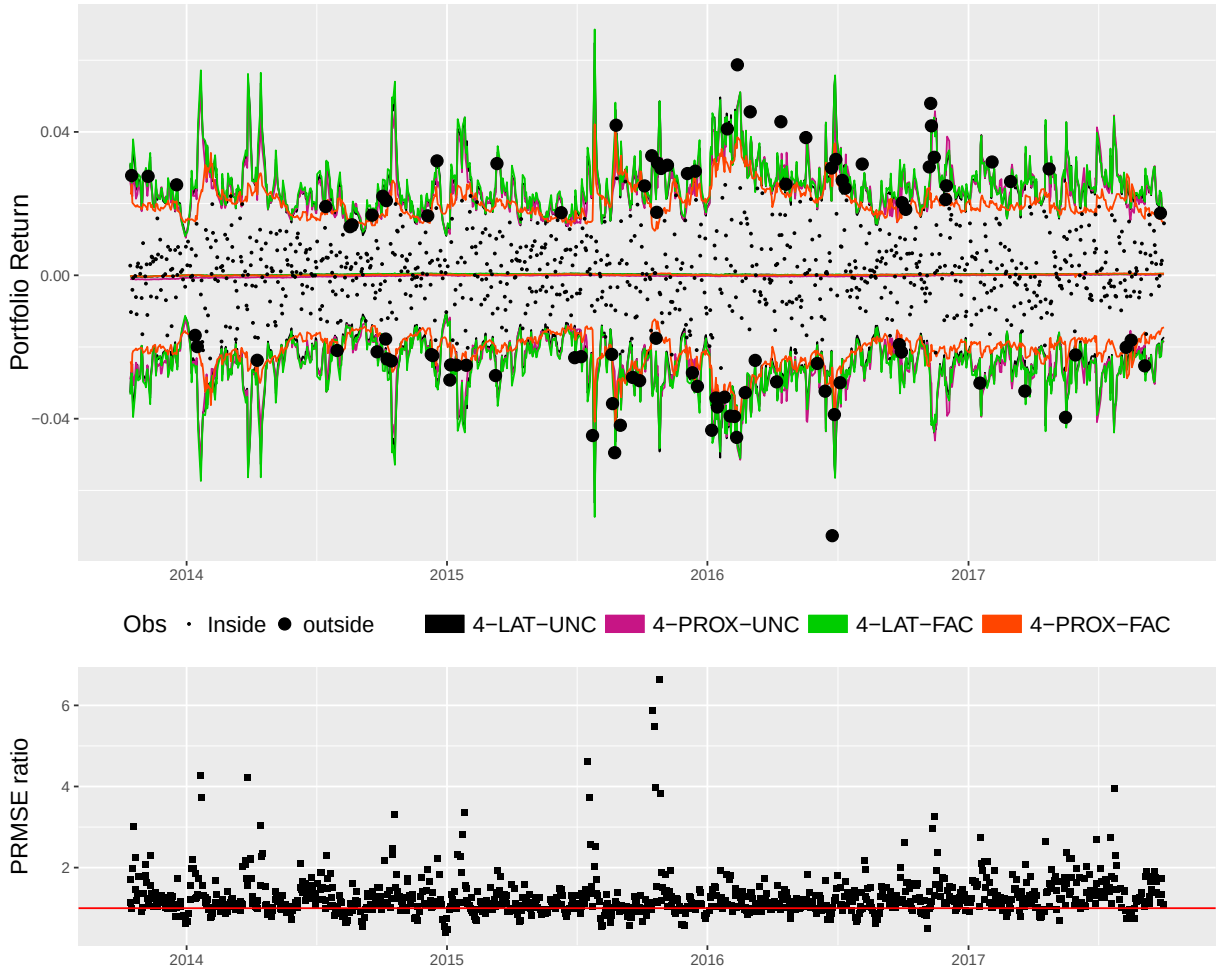


Figure 3.8: *Top:* Posterior means and 95% prediction intervals for the equally-weighted portfolio return forecasts of each model. True return values are denoted by circles. *Bottom:* Ratio of 4-LAT-UNC model PRMSEs to those of 4-PROX-FAC.

October 2015 and September 2017, with parameters calibrated on the $W = 1500$ training days proceeding each test day.

Among many possible approaches to American option pricing (e.g., Cox et al., 1979; Tilley, 1993; Fu et al., 2001; Rogers, 2002; Ibanez and Zapatero, 2004), here we adopt the celebrated method of Longstaff and Schwartz (2001). The method consists of simulating

multiple paths of S_t at daily resolution on the time interval $(t, T]$ in the risk-neutral measure. Starting from the known payoff of the option at time T (for a Call option this is $\min\{S_T - K, 0\}$), the method recursively works backwards to approximate the conditional payoff at earlier times by simple least-squares. At the final step, we obtain the option price O_t upon adjusting for the risk-free rate.

Due to the market is incomplete under the SV models, the risk neutral measure is not unique. For the single-asset eOU model, the risk-neutral measure suggested by Perelló et al. (2008) under minimal risk aversion takes the form

$$\begin{aligned} \mathrm{d}S_t &= rS_t\,\mathrm{d}t + S_tV_t\,\mathrm{d}W_t^S, \\ \mathrm{d}\log V_t &= \gamma(\mu - \log V_t)\,\mathrm{d}t + \sigma\,\mathrm{d}W_t^V, \qquad \mathrm{cor}(\mathrm{d}W_t^S, \mathrm{d}W_t^V) = \rho, \end{aligned} \tag{3.15}$$

where r is the risk-free rate, W_t^S and W_t^V are Brownian motions in the risk-neutral measure, and γ, μ, σ, and ρ are the same eOU parameters as in the physical measure. Since these parameters are unknown and to be estimated from historical data, we have the following Bayesian method 3.2 for determining the price O_n of an American-style JPM option contract on day n with an eOU-mSV model:

Algorithm 3.2: Bayesian option price simulation

1 Suppose we employ W past days to train the mSV model, such that the training data is $\boldsymbol{Y}_n^W = (\boldsymbol{Y}_{n-W+1}, \ldots, \boldsymbol{Y}_n)$ as defined in (3.14).

2 Use MCMC to sample M draws from the posterior distribution $p(V_{\mathrm{JPM},n}, \boldsymbol{\theta}|\boldsymbol{Y}_n^W)$, where $V_{\mathrm{JPM},n}$ is the latent JPM volatility on day n.

3 For each draw $(V_{\mathrm{JPM},n}^{(m)}, \boldsymbol{\theta}^{(m)})$ obtained in Step 2, calculate the corresponding $O_n^{(m)} = \mathcal{G}(S_{\mathrm{JPM},n}, V_{\mathrm{JPM},n}^{(m)}, \boldsymbol{\theta}^{(m)}, r, K, T)$ using the method of Longstaff and Schwartz (2001), with paths simulated from the risk-neutral version (3.15) of the marginal eOU model for JPM. The collection of all M of these values is a sample from the posterior option price distribution $p(O_n|\boldsymbol{Y}_n^W)$.

Figure 3.9 displays the posterior option price distribution for several Call/Put contracts valued on May 9, 2017 at risk-free rate $r = 0$, with different eOU-MSV models at different

levels of moneyness

$$\mathcal{M}_n(K) = \begin{cases} S_{\text{JPM},n} - K, & \text{Call option,} \\ K - S_{\text{JPM},n}, & \text{Put option.} \end{cases}$$

For most contracts, the posterior distributions obtained with VIX proxy (PROX) are much

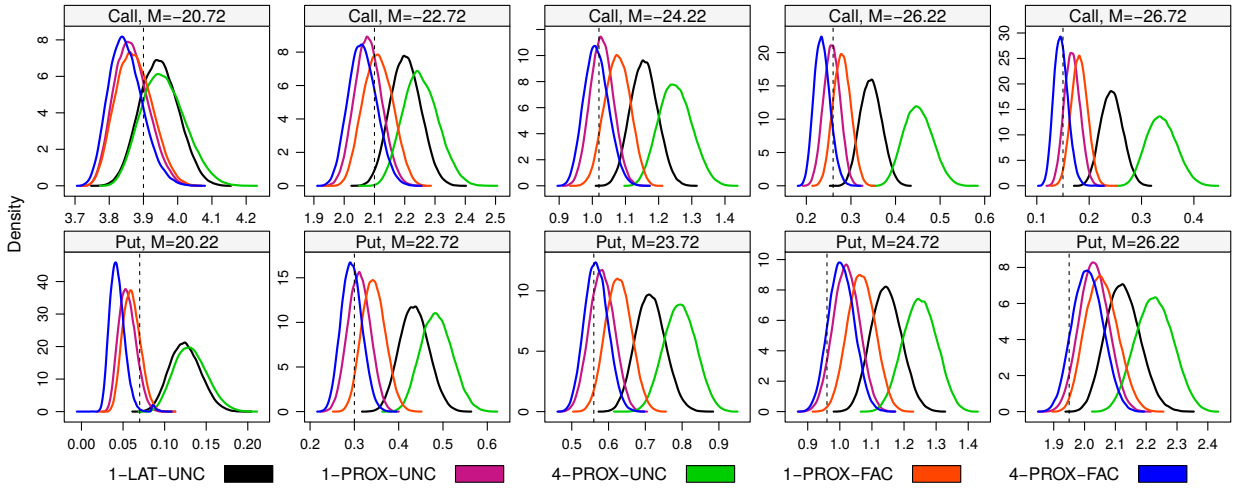


Figure 3.9: Posterior distribution of JPM option prices on May 9, 2017, for various models and level of moneyness $\mathcal{M}_n(K)$. The time to maturity is 8 days. Dotted lines denote the traded option price.

closer to the traded option value that those without (LAT).

Figure 3.10 compares the RMSEs

$$\text{RMSE}_n(K) = \sqrt{E\big[\{O_n(K) - O_n^{\text{obs}}(K)\}^2 | \boldsymbol{Y}_n^W\big]}$$

of various option price contracts over the $N_{\text{test}} = 500$ testing days in our study. Contracts were grouped by type (Call/Put), days to maturity $T - n$, and three levels of moneyness as shown in Table 3.6. For assessing our mSV models' option pricing performance, we only considered contracts having a trade volume of at least 50 (each unit is written on 100 shares of stocks) on a given day. However, arguably the major utility of the posterior distribution $p(O_n|\boldsymbol{Y}_n^W)$ is to price more thinly traded options, for which the market price may be unreliable or even unavailable. To facilitate comparisons between groups, the

Option Type	In the Money	At the Money	Out of the Money
Call Option	90-98 %	98-102 %	102-110 %
Put Option	102-110 %	98-102 %	90-98 %

Table 3.6: Classification of options by moneyness, defined here as $S_{\text{JPM},n}/K$.

RMSEs in Figure 3.10 are standardized by the average traded value of the options in each group.

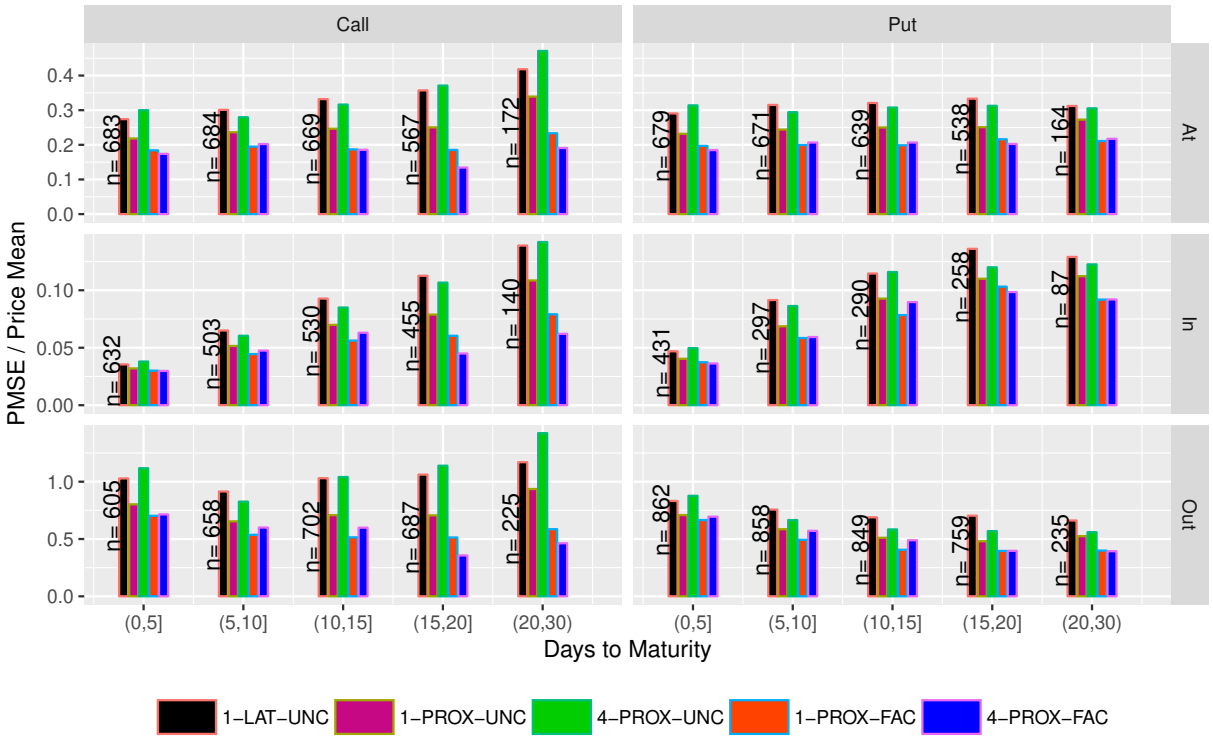


Figure 3.10: Standardized RMSEs of JPM option by type (Call/Put), moneyness (At/In/Out) and days to maturity $T - n$. The number of contracts in each group is indicated on the left of its bars.

In all groups, the models with common-factor correlation structure (FAC) have the lowest RMSE, confirming the findings in Figure 3.9. For Call options where the time-

to-maturity is more than 15 days, multi-asset modeling (4-PROX-FAC) produces better pricing results than modeling JPM alone (1-PROX-FAC). However, for Put options at 15-day maturity, multi-asset modeling seems to have little effect.

3.4.3 Impact of Correlation Matrix Design

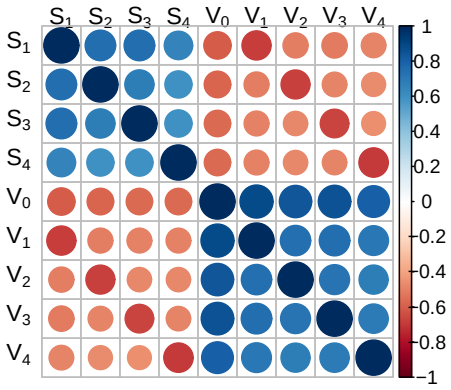
In the empirical assessments of Sections 3.4.1 and 3.4.2, we found that our proposed multi-asset model with common-factor volatility proxy (4-PROX-FAC) had much narrower volatility estimates than the unconstrained correlation matrix UNC. This translated to considerably better option pricing RMSEs, but somewhat more forecasting undercoverage of the true returns at nominal values of $\alpha \geq 80\%$. In this section we attempt to elucidate these findings.

Figure 3.11 compares the posterior mean of the 4-PROX-FAC, 4-PROX-UNC, 4-LAT-UNC and 4-LAT-FAC correlation matrices fitted to the $q = 4$ financial assets JPM, C, GS, and COF for $N_{\text{train}} = 1500$ training days between Oct. 26, 2017 and Oct. 10, 2013. All of the models estimate the same correlation structure within the Brownian innovations of the assets ($\text{cor}(\boldsymbol{B}_t^S)$) and within those of the volatilities ($\text{cor}(\boldsymbol{B}_t^V)$). However, it is perhaps surprising that only the FAC design with the VIX observed forces considerable negative correlations between a given asset and its instantaneous volatility, whereas the other three models allow these correlations to be essentially zero.

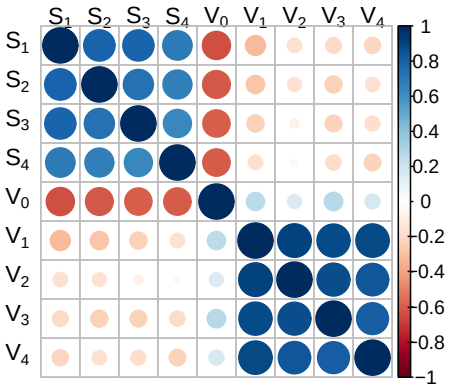
The explanation for this finding is as follows. Since both correlation estimates of PROX-FAC and PROX-UNC condition on the VIX as a volatility proxy, there is considerable information in the data to estimate both within-asset correlations, $\text{cor}(\boldsymbol{B}_t^S)$, and correlations between the asset and the proxy, $\text{cor}(\boldsymbol{B}_t^S, B_{0t}^V)$. Indeed, the top left corners of Figure 3.11a and Figure 3.11b are the same. However, in (3.8) we have shown that for the common-factor mSV model,

$$\text{cor}(B_{0t}^V, B_{it}^V) = \tau_i, \qquad \text{cor}(B_{it}^V, B_{it}^S) = \rho_i, \qquad \text{cor}(B_{0t}^V, B_{it}^S) = \rho_i \tau_i,$$

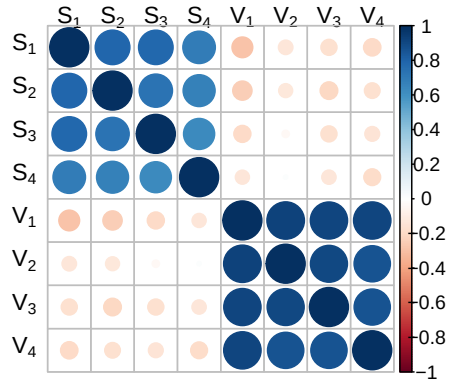
such that the only way to capture the strong correlation between assets and the VIX exhibited by the empirical data is for both $|\rho_i|$ and $|\tau_i|$ to be large. Hence when the VIX is



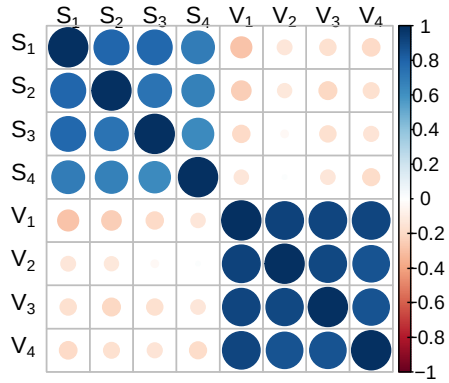
(a) 4-PROX-FAC



(b) 4-PROX-UNC



(c) 4-LAT-FAC



(d) 4-LAT-UNC

Figure 3.11: Posterior mean of 4-asset correlation matrices for common-factor and unconstrained correlation designs.

observed, the FAC structure results in a strong negative correlation between the asset and its volatility, or leverage effect. On the other hand, in the UNC correlation structure with the VIX observed the latent volatilities are free to simply capture the magnitude of asset fluctuations, since the leverage effect is already accounted for by the VIX. In Figure 3.11c and Figure 3.11d, we compare the two models with the VIX unobserved. Both of the models choose to have a similar structure as the PROX-UNC model. It means that the FAC design only makes a difference when the proxy is observed.

By design, the UNC correlation structure has more flexibility to fit the data; in our experiment, the extra flexibility makes better forecasts (less undercoverage for extreme values in Tables 3.4 and 3.5). However, by forcing the individual volatilities to exhibit the leverage effect, the PROX-FAC design estimates V_{in} more like the *implied* volatility, which has long been noted to have strong negative correlation with the asset price (Figlewski and Wang, 2000; Bouchaud et al., 2001). For this reason, the PROX-FAC model is found to be the better model for option pricing.

3.5 Conclusion

In this chapter, we present a multi-asset stochastic volatility (mSV) modeling framework with common-factor volatility. The parsimonious factor design allows the number of model parameters to scale linearly with the number of assets, while remaining sufficiently flexible to make comparable forecasts to a model with unconstrained correlation structure. Furthermore, it is shown that substituting an observable proxy (such as VIX) for the common volatility factor can greatly improve the precision of parameter and latent volatility estimates. Combined with the common-factor structure, this latter property translates to generally equivalent forecasting results and better option pricing results compared to several competing multi-asset models. Note that the model parameters are estimated only by the asset prices instead of being calibrated by the option prices, it suggests that the proposed common-factor model with proxy is a useful tool for pricing illiquid options.

There exist limitations in our study and those limitations would benefit from further research. The common volatility proxy we used is the VIX which is essentially an implied

volatility. We believe that it is the reason of our model's advantages on option pricing. However, replacing it with a realized volatility proxy might benefit the proposed model's performance on forecasting. Limited by the computational power, our data analysis is based on 1000 results which can be considerably expanded. The assets chosen for our study are financial assets whose comovements with the SPX are quite clear. The option pricing study can be extended to other assets and has larger selection of options, e.g., longer maturity. It will be an interesting topic to include some technology companies and see whether the PROX-FAC model can be useful. As more assets are included, the possibility of existing more than one factor among the asset volatilities increase, hence, we might need to find proxies for all the factors.

Chapter 4

The Synthetic Kalman Filter: A Fast Filtering Method for Conditionally Gaussian State-Space Models

4.1 Introduction

In the previous chapter, we proposed a multi-asset SV model based on the stochastic differential equations (SDEs) of the form

$$\mathrm{d}\boldsymbol{Y}_t = \boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t)\Delta\mathrm{T} + \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{1/2}(\boldsymbol{Y}_t)\mathrm{d}\boldsymbol{B}_t, \tag{4.1}$$

where $\boldsymbol{\theta}$ represents the model parameters, $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{1/2}(\boldsymbol{Y}_t)$ is the lower-triangular Cholesky factor of $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t)$, $\boldsymbol{B}_t$ is a vector of independent Brownian motions and the model variable vector $\boldsymbol{Y}_t$ is partially observed, i.e., $\boldsymbol{Y}_t = (\boldsymbol{X}_t, \boldsymbol{Z}_t)$, where $\boldsymbol{X}_t$ are observed variable vector of dimension p and $\boldsymbol{Z}_t$ are latent variable vector of dimension q.

To obtain the likelihood of the SDE (4.1), an Euler approximation is often used in practice. Suppose that $\boldsymbol{X}_{1:n} = (\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n)$ are data recorded with interobservation time $\Delta\mathrm{T}$. As we shall operate in discrete time for the remainder of this chapter, by abuse of notation let $\boldsymbol{X}_t, t = 1, \cdots, n$ denote the tth observation, with similar notations

for $\boldsymbol{Y}_t$, $\boldsymbol{Z}_t$, $\boldsymbol{Y}_{1:n} = (\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n)$, and $\boldsymbol{Z}_{1:n} = (\boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_n)$. Then the Euler-Maruyama approximation (Maruyama, 1955) for the intractable transition density of the SDE (4.1) is:

$$\boldsymbol{Y}_{t+1} = \boldsymbol{\Lambda}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t) + \boldsymbol{\Omega}_{\boldsymbol{\theta}}^{1/2}(\boldsymbol{Y}_t)\boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \overset{\text{iid}}{\sim} \mathcal{N}_{p+q}(\boldsymbol{0}, \boldsymbol{I}), \tag{4.2}$$

where $\boldsymbol{\Lambda}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t) = \boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t) \times \Delta\mathrm{T} + \boldsymbol{Y}_t$, $\boldsymbol{\Omega}_{\boldsymbol{\theta}}^{1/2}(\boldsymbol{Y}_t)$ is the lower-triangular Cholesky factor of $\boldsymbol{\Omega}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t)$, where $\boldsymbol{\Omega}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t) = \boldsymbol{\Sigma}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t) \times \Delta\mathrm{T}$. We refer to this model (4.2) as a generalized Gaussian state-space model (GG-SSM), in the sense that the transition densities are Gaussian, but the means and variances of $\boldsymbol{X}_t$ and $\boldsymbol{Z}_t$ can be nonlinear functions of $\boldsymbol{X}_{t-1}$ and $\boldsymbol{Z}_{t-1}$.

GG-SSMs of the form (4.2) provide a versatile and expressive modeling framework for time series dynamics. However, parameter inference is complicated by the fact that the likelihood $\mathcal{L}(\boldsymbol{X}_{1:n}|\boldsymbol{\theta})$ is not available in closed-form. That is, the observed data likelihood is given by

$$\mathcal{L}(\boldsymbol{X}_{1:n}|\boldsymbol{\theta}) = p(\boldsymbol{X}_{1:n}|\boldsymbol{\theta}) = \int \prod_{t=2}^{n} p_{\boldsymbol{\theta}}(\boldsymbol{Y}_t|\boldsymbol{Y}_{1:t-1}) \mathrm{d}\boldsymbol{Z}_{1:n}, \tag{4.3}$$

where $p_{\boldsymbol{\theta}}(\boldsymbol{Y}_t|\boldsymbol{Y}_{t-1})$ is the transition density of the Euler-Maruyama approximation (4.2), and the integral must be performed by numerical methods. One such method is by Bayesian data augmentation. That is, for given prior $\pi(\boldsymbol{\theta})$, samples are drawn from the augmented distribution

$$p(\boldsymbol{Z}_{1:n}, \boldsymbol{\theta}|\boldsymbol{X}_{1:n}) \propto \pi(\boldsymbol{\theta}) p(\boldsymbol{Y}_{1:n}|\boldsymbol{\theta}) = \pi(\boldsymbol{\theta}) \prod_{t=2}^{n} p_{\boldsymbol{\theta}}(\boldsymbol{Y}_t|\boldsymbol{Y}_{t-1}), \tag{4.4}$$

typically using Markov chain Monte Carlo (MCMC) methods (e.g., Jacquier et al., 1994; Eraker, 2001; Broto and Ruiz, 2004; Kou et al., 2012; Beskos et al., 2013; Bladt et al., 2016; Kastner et al., 2017). In Chapter 3, we relied heavily on such a fully Bayesian MCMC method implemented by the R package *msde* Lysy and Tong (2017) to analyze the multi-asset stochastic volatility models. The details of this algorithm can be found in Appendix B. However, since parameters and latent variables in (4.4) typically are highly correlated with each other, the mixing time of this MCMC algorithm tends to be very slow.

Another approach to parameter estimation is to sequentially filter out the latent variables, rather than integrate them out all at once. That is, the likelihood can be factored

as

$$\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n}) = \prod_{t=2}^{n} p_{\boldsymbol{\theta}}(\boldsymbol{X}_t|\boldsymbol{X}_{1:t-1}), \tag{4.5}$$

such that if the filtering density $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t-1}|\boldsymbol{X}_{1:t-1})$, were given, we could use it to calculate both

$$\text{(Prediction)} \qquad p_{\boldsymbol{\theta}}(\boldsymbol{X}_t|\boldsymbol{X}_{1:t-1}) = \int p_{\boldsymbol{\theta}}(\boldsymbol{X}_t|\boldsymbol{X}_{t-1}, \boldsymbol{Z}_{t-1}) p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t-1}|\boldsymbol{X}_{1:t-1}) d\boldsymbol{Z}_{t-1}, \tag{4.6}$$

$$\begin{aligned} \text{(Update)} \qquad p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t}) &= \int p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t, \boldsymbol{Z}_{t-1}|\boldsymbol{X}_{1:t}) d\boldsymbol{Z}_{t-1} \\ &= \frac{\int p_{\boldsymbol{\theta}}(\boldsymbol{X}_t, \boldsymbol{Z}_t|\boldsymbol{X}_{t-1}, \boldsymbol{Z}_{t-1}) p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t-1}|\boldsymbol{X}_{1:t-1}) d\boldsymbol{Z}_{t-1}}{p_{\boldsymbol{\theta}}(\boldsymbol{X}_t|\boldsymbol{X}_{1:t-1})}. \end{aligned} \tag{4.7}$$

where the *prediction* equation (4.6) is used to evaluate the likelihood in (4.5), and the *update* equation (4.7) determines the next filtering density from the previous one. Thus, instead of performing one $N \times q$ dimensional integral to obtain the observed data likelihood $\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$ (4.3), the filtering approach divides the problem into N consecutive steps involving just q-dimensional integrals for each of $p_{\boldsymbol{\theta}}(\boldsymbol{X}_t|\boldsymbol{X}_{1:t-1})$ and $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t})$.

For a linear GG-SSM with the form of

$$\boldsymbol{\Lambda}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t) = \boldsymbol{\Lambda}_{\boldsymbol{\theta}}\boldsymbol{Y}_t + \boldsymbol{\Gamma}_{\boldsymbol{\theta}}, \quad \boldsymbol{\Omega}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t) = \boldsymbol{\Omega}_{\boldsymbol{\theta}}, \tag{4.8}$$

all conditional densities of the model are Gaussian, so that the filter only needs to update the mean and variance of $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t})$. By properties of the multivariate Gaussian distribution, the updates can be performed exactly, leading to the celebrated Kalman filter (Kalman et al., 1960).

For the general, nonlinear GG-SSM (4.2), several extensions of the Kalman filter have been proposed to conduct approximate parameter inference. Typically, the filtering density is assumed to be Gaussian or a mixture of Gaussians, with different method of performing the filtering step. Some methods obtain the filtering mean and variance deterministically, e.g., using Taylor expansions (Anderson and Moore, 1979) or from a lattice of optimal evaluation points (Julier and Uhlmann, 1997; Arasaratnam and Haykin, 2009). Other Kalman-type variants filter information stochastically, for example the Monte Carlo

Kalman filter (Song, 2000) and the Ensemble Kalman filter (Evensen, 1994; Katzfuss et al., 2016), the latter having been adapted to state-space models which are not conditionally Gaussian (e.g., Stordal et al., 2011; Hoteit et al., 2012).

Originally proposed by Gordon et al. (1993), particle filtering (PF) methods can represent an arbitrary filtering distribution $p_{\boldsymbol{\theta}}\boldsymbol{Z}_t|\boldsymbol{X}_{1:t}$ via a weighted sample of particles. These methods can be used to conduct exact parameter inference for highly nonlinear GG-SSMs. In fact, particle filters can be used to construct so-called *pseudo-marginal* MCMC algorithms directly on the parameter space, for which the stationary distribution is exactly the posterior density $p(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$ Andrieu et al. (2009, 2010). However, particle filters are usually quite computationally intensive, since the required number of particles increases exponentially with both the number of latent dimensions and the number of observations (Snyder et al., 2008; Bengtsson et al., 2008; Doucet and Johansen, 2009). In fact, for mildly nonlinear problems, several authors have reported inferior performance to various Kalman-type variants (e.g., Weerts and El Serafy, 2006; Han and Li, 2008; Särkkä et al., 2015).

In this chapter, we propose a new stochastic variant on the Kalman filter, which we term Synthetic Kalman Filter (SKF) by analogy with the synthetic likelihoods of Wood (2010); Price et al. (2017). The SKF is tailored to moderately nonlinear, GG-SSMs, such as the multi-asset stochastic volatility (mSV) models proposed in the previous chapter. By approximating both $p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t})$ and $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t})$ as Gaussian, an efficient importance sampling algorithm is employed to filter only means and variances, which requires far fewer particles than for an entire distribution. Extensive simulation results indicate that, for mSV models containing up to 5000 latent variables, the SKF approximation incurs little loss of accuracy relative to exact inference methods; however, the computational burden can be decreased by an order of magnitude or more.

The remainder of this chapter is organized as follows. In Section 4.2, we revisit the Kalman filter and the particle filter for the GG-SSM (4.2). Section 4.3 presents the SKF method. Section 4.4 discusses various computational aspects of the Bayesian implementation of the SKF. Numerical evidence will be presented in Section 4.5 to assess the Gaussian assumption in the SKF method. Section 4.6 compares the SKF method to competing state-of-the-art methods of parameter inference for the multi-asset SV model (4.16) proposed in

Chapter 3. We summarize our findings in Section 4.7.

4.2 Filtering Methods for GG-SSMs

In this section we review the Kalman and particle filters for the GG-SSM (4.2), noting that they are typically presented for slightly simpler state-space models in which $\boldsymbol{X}_t$ and $\boldsymbol{Z}_t$ are conditionally independent given $\boldsymbol{Z}_{t-1}$. For ease of presentation, we decompose the mean function $\boldsymbol{\Lambda}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t)$, the variance-covariance function $\boldsymbol{\Omega}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t)$ and the random increments $\boldsymbol{\epsilon}_t$ of the GG-SSM (4.2) into the parts corresponding to the observed $\boldsymbol{X}_t$ and latent $\boldsymbol{Z}_t$, such that

$$\boldsymbol{\Lambda}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t) = \begin{matrix} \\ p \\ q \end{matrix} \begin{matrix} 1 \\ \begin{pmatrix} \boldsymbol{\Lambda}_t^{\mathbf{X}} \\ \boldsymbol{\Lambda}_t^{\mathbf{Z}} \end{pmatrix} \end{matrix}, \quad \boldsymbol{\Omega}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t) = \begin{matrix} \\ p \\ q \end{matrix} \begin{matrix} p & q \\ \begin{pmatrix} \boldsymbol{\Omega}_t^{\mathbf{XX}} & \boldsymbol{\Omega}_t^{\mathbf{XZ}} \\ \boldsymbol{\Omega}_t^{\mathbf{ZX}} & \boldsymbol{\Omega}_t^{\mathbf{ZZ}} \end{pmatrix} \end{matrix}, \quad \boldsymbol{\epsilon}_t = \begin{matrix} \\ p \\ q \end{matrix} \begin{matrix} 1 \\ \begin{pmatrix} \boldsymbol{\epsilon}_{\mathbf{X}} \\ \boldsymbol{\epsilon}_{\mathbf{Z}} \end{pmatrix} \end{matrix}, \tag{4.9}$$

where $\boldsymbol{\Lambda}_t^{\mathbf{X}}$, $\boldsymbol{\Lambda}_t^{\mathbf{Z}}$, $\boldsymbol{\Omega}_t^{\mathbf{XX}}$, $\boldsymbol{\Omega}_t^{\mathbf{XZ}}$ and $\boldsymbol{\Omega}_t^{\mathbf{ZZ}}$ are functions of $\boldsymbol{Y}_t$ and $\boldsymbol{\theta}$.

4.2.1 Kalman Filter

The original Kalman filter was proposed for the linear model (4.8), which has a linear drift term and constant variance-covariance matrix. All of the relevant densities are Gaussian due to the linear nature of the model. The procedures of the Kalman filter are described in Algorithm 4.1.

Note that we can write the updated filtering density of GG-SSM (4.2) by (4.7) For the linear model (4.8), both $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}, \boldsymbol{X}_{t+1}|\boldsymbol{Z}_t, \boldsymbol{X}_t)$ and $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t})$ follow Gaussian distributions, in which the means are linear on $\boldsymbol{Z}_t$ and $\boldsymbol{X}_t$ and the variance matrices are constant. As a result, $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}|\boldsymbol{X}_{1:t+1})$ is also Gaussian. Hence the Kalman filter bypasses this integral by finding the exact mean and variance directly.

For nonlinear models, several variants of the Kalman filter have been proposed, e.g., Anderson and Moore (1979); Julier and Uhlmann (1997); Arasaratnam and Haykin (2009). They usually follow the *prediction* and *updating* procedures as in Algorithm 4.1. However, these extensions can only produce approximate likelihood inference, each with a different

Algorithm 4.1: Kalman Filter

Input : $\boldsymbol{\theta}$; $\boldsymbol{Z}_1|\boldsymbol{X}_1, \boldsymbol{\theta} \sim \mathcal{N}_q(\boldsymbol{\lambda}_1, \boldsymbol{\Psi}_1)$

Output: $\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$

1 Start the algorithm with $t = 1$ that $\boldsymbol{Z}_t|\boldsymbol{X}_{1:t}, \boldsymbol{\theta} \sim \mathcal{N}_q(\boldsymbol{\lambda}_t, \boldsymbol{\Psi}_t)$.

2 *Prediction step.* Due to the Gaussian property, we can write the distribution of $\boldsymbol{Y}_{t+1}$ given the previous observed information as

$$\boldsymbol{Y}_{t+1}|\boldsymbol{X}_{1:t}, \boldsymbol{\theta} \sim \mathcal{N}_{p+q}\left(\boldsymbol{\Lambda}_{\boldsymbol{\theta}}(\hat{\boldsymbol{Y}}_t), \boldsymbol{\Omega}_{\boldsymbol{\theta}}(\hat{\boldsymbol{Y}}_t) + \boldsymbol{\Lambda}\boldsymbol{\Psi}_t\boldsymbol{\Lambda}'\right)$$

where $\hat{\boldsymbol{Y}}_t = (\boldsymbol{X}_t, \boldsymbol{\lambda}_t)$.

3 *Update step.* By including the information of $\boldsymbol{X}_{t+1}$, we can find the conditional Gaussian distribution of

$$\boldsymbol{Z}_{t+1}|\boldsymbol{X}_{1:t+1}, \boldsymbol{\theta} \sim \mathcal{N}_q\left(\hat{\boldsymbol{\Lambda}}_{\boldsymbol{t}}^{\mathbf{Z}} + \boldsymbol{K}_t(\boldsymbol{X}_{t+1} - \hat{\boldsymbol{\Lambda}}_{\boldsymbol{t}}^{\mathbf{X}}), \hat{\boldsymbol{\Omega}}_{\boldsymbol{t}}^{\mathbf{ZZ}} - \boldsymbol{K}_t\hat{\boldsymbol{\Omega}}_{\boldsymbol{t}}^{\mathbf{XZ}}\right)$$

where $\boldsymbol{\Lambda}_{\boldsymbol{\theta}}(\hat{\boldsymbol{Y}}_t) = \begin{pmatrix} \hat{\boldsymbol{\Lambda}}_{\boldsymbol{t}}^{\mathbf{X}} \\ \hat{\boldsymbol{\Lambda}}_{\boldsymbol{t}}^{\mathbf{Z}} \end{pmatrix}$, $\boldsymbol{\Omega}_{\boldsymbol{\theta}}(\hat{\boldsymbol{Y}}_t) + \boldsymbol{\Lambda}\boldsymbol{\Psi}_t\boldsymbol{\Lambda}' = \begin{pmatrix} \hat{\boldsymbol{\Omega}}_{\boldsymbol{t}}^{\mathbf{XX}} & \hat{\boldsymbol{\Omega}}_{\boldsymbol{t}}^{\mathbf{XZ}} \\ \hat{\boldsymbol{\Omega}}_{\boldsymbol{t}}^{\mathbf{ZX}} & \hat{\boldsymbol{\Omega}}_{\boldsymbol{t}}^{\mathbf{ZZ}} \end{pmatrix}$ and $\boldsymbol{K}_t = \hat{\boldsymbol{\Omega}}_{\boldsymbol{t}}^{\boldsymbol{ZX}}\left(\hat{\boldsymbol{\Omega}}_{\boldsymbol{t}}^{\mathbf{XX}}\right)^{-1}$.

4 For the model likelihood, we evaluate the marginal transition density of $p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t})$, the marginal of the joint Gaussian distribution $p_{\boldsymbol{\theta}}(\boldsymbol{Y}_{t+1}|\boldsymbol{X}_{1:t})$ in step 2.

5 Given the updated filtering density $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}|\boldsymbol{X}_{1:t+1})$ in step 3, repeat step 2-4 by replacing t with $t + 1$.

6 Collect $p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t})$, $t = 1, \ldots, n-1$, so that $\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$ can be calculated by (4.3).

tradeoff between speed and accuracy. We shall revisit some of these extensions in more detail in the numerical comparisons of Section 4.6.1.

4.2.2 Particle Filter

For a generic model setting where the mean and variance functions in (4.2) can be nonlinear and depend on the model variables, the original Kalman filter is not suitable for two reasons. The first is that the filtering density $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t})$ is no longer Gaussian, and can be analytically unavailable. The second is that the transition density $p_{\boldsymbol{\theta}}(\boldsymbol{Y}_{t+1}|\boldsymbol{Y}_t)$ is always Gaussian due to the model specification, but the mean of this Gaussian distribution is not linear on $\boldsymbol{Y}_t$ and the variance-covariance matrix can depend on $\boldsymbol{Y}_t$. Hence, the analytical Gaussian result that allows us to bypass the integral (4.7) is no longer available.

The particle filter is an exact method to solve such a nonlinear system. The sequential importance resampling (SIR) scheme is one popular implementation of the particle filter (see examples in Liu and Chen, 1998; Doucet et al., 2000; Andrieu and Doucet, 2002). We introduce the specific sequential importance resampling technique used by Andrieu et al. (2010) in Algorithm 4.2.

As shown in Algorithm 4.2, the particle filter constructs an importance sampler for each $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{1:t}|\boldsymbol{X}_{1:t})$ iteratively. In each iteration, once the weighted sample $\left[\boldsymbol{Z}_{1:t}^{(i)}, w_t^{(i)}\right]_{i=1,\ldots,M}$, has been obtained, we first need to resample from it so that $\left[\boldsymbol{Z}_{1:t}^{(i)*}\right]_{i=1,\cdots,M}$ is created. Next, the sample $\left[\boldsymbol{Z}_{1:t}^{(i)*}\right]_{i=1,\cdots,M}$ is updated into an importance sampler for $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{1:t+1}|\boldsymbol{X}_{1:t+1})$ by

1. drawing $\boldsymbol{Z}_{t+1}^{(i)*} \sim q_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}|\boldsymbol{Z}_{1:t}^{(i)*})$, where $q_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}|\boldsymbol{Z}_{1:t}^{(i)*}) = p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}|\boldsymbol{X}_t, \boldsymbol{Z}_t^{(t)*})$;
2. setting the unnormalized weight $w_{t+1}^{(i)}$ of $\boldsymbol{Z}_{1:t+1}^{(i)*}$ by

$$p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{Z}_{t+1}^{(i)*}, \boldsymbol{Z}_t^{(i)*}, \boldsymbol{X}_t) \propto \frac{p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{1:t+1}^{(i)*}|\boldsymbol{X}_{1:t+1})}{p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{1:t}^{(i)*}|\boldsymbol{X}_{1:t})p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}^{(i)*}|\boldsymbol{X}_t, \boldsymbol{Z}_t^{(i)*})}$$

.

After we obtain the normalized weights $\left[w_{t+1}^{(i)*}\right]_{i=1,\ldots,M}$, we collect the new set of $\left[\boldsymbol{Z}_{1:t+1}^{(i)}, w_{t+1}^{(i)}\right]_{i=1,\ldots,M}$ as shown in step 6 of Algorithm 4.2. Then, we can enter the next iteration.

Note that $\boldsymbol{Z}_{1:t+1}^{(i)*}$ are sampled from $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{1:t}|\boldsymbol{X}_{1:t})p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}|\boldsymbol{X}_t, \boldsymbol{Z}_t)$ and the weights (4.10) are the realizations of $\boldsymbol{Z}_{1:t+1}^{(i)*}$, $i = 1, \ldots, M$, with probability $p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{Z}_{t+1}, \boldsymbol{Z}_t, \boldsymbol{X}_t)$. Hence

Algorithm 4.2: Sequential Importance Resampling

Input : $\boldsymbol{\theta}$; $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_1|\boldsymbol{X}_1)$

Output: $\hat{\mathcal{L}}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$

1 Start the algorithm with $t = 1$ and a sample of $\left[\boldsymbol{Z}_{1:t}^{(i)*}\right]_{i=1,\ldots,M}$ from $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{1:t}|\boldsymbol{X}_{1:t})$.

2 Sample $\boldsymbol{Z}_{t+1}^{(i)*}$ from the marginal transition density of $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}|\boldsymbol{X}_t, \boldsymbol{Z}_t^{(i)*})$, which forms a path $\boldsymbol{Z}_{1:t+1}^{(i)*}$, for $i = 1, \ldots, M$.

3 Compute the weights

$$\omega_{t+1}^{(i)} = p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{Z}_{t+1}^{(i)*}, \boldsymbol{Z}_t^{(i)*}, \boldsymbol{X}_t), \tag{4.10}$$

which is the conditional Gaussian of $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}^{(i)}, \boldsymbol{X}_{t+1}|\boldsymbol{X}_t, \boldsymbol{Z}_t^{(i)})$.

4 Normalize the weights

$$\omega_{t+1}^{(i)*} = \frac{\omega_{t+1}^{(i)}}{\sum_{j=1}^{M} \omega_{t+1}^{(j)}}, \quad , i = 1, \ldots, M.$$

5 Note that we can estimate the marginal transition density $p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t})$ by a average of $\left[\omega_{t+1}^{(i)}\right]_{i=1,\ldots,M}$ such that

$$\hat{p}_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t}) = \frac{1}{M}\sum_{i=1}^{M} \omega_{t+1}^{(i)}.$$

6 Define the notation $\left[\boldsymbol{Z}_{1:t+1}^{(i)}, \omega_{t+1}^{(i)}\right]_{i=1,\ldots,M}$ by copying $\left[\boldsymbol{Z}_{1:t+1}^{(i)*}, \omega_{t+1}^{(i)*}\right]_{i=1,\ldots,M}$, so that $\left[\boldsymbol{Z}_{1:t+1}^{(i)}, \omega_{t+1}^{(i)}\right]_{i=1,\ldots,M}$ is a weighted sample representing $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{1:t+1}|\boldsymbol{X}_{1:t+1})$. Reset $\left[\boldsymbol{Z}_{1:t+1}^{(i)*}\right]_{i=1,\ldots,M}$ by a sample of size M from $\left[\boldsymbol{Z}_{1:t+1}^{(i)}, \omega_{t+1}^{(i)}\right]_{i=1,\ldots,M}$, then we repeat step 2-5 by replacing t with $t + 1$ until $t = n - 1$.

7 Collect $\hat{p}_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t})$, $t = 1, \ldots, n-1$, so that $\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$ can be estimated by (4.5).

the average of the weights $\left[\omega_{t+1}^{(i)}\right]_{i=1,\ldots,M}$ is an Monte Carlo estimator of the integral of

$$p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t}) = \int p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{Z}_{t+1}, \boldsymbol{Z}_t, \boldsymbol{X}_t) p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{1:t}|\boldsymbol{X}_{1:t}) p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}|\boldsymbol{X}_t, \boldsymbol{Z}_t) d\boldsymbol{Z}_{1:t+1}.$$

Hence, the model likelihood can be estimated by the product of those estimated marginal transition densities by (4.5).

In order to perform Bayesian analysis on the parameters by particle filter, we can apply the pseudo-marginal MCMC approach. That is, to draw samples from $p(\boldsymbol{\theta}, \boldsymbol{Z}_{1:n}|\boldsymbol{X}_{1:n}) = p(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})p(\boldsymbol{Z}_{1:n}|\boldsymbol{\theta}, \boldsymbol{X}_{1:n})$. Note $\hat{\mathcal{L}}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$ produced by Algorithm 4.2 is in fact a positive unbiased estimate of $\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$ and we can draw from $p(\boldsymbol{Z}_{1:n}|\boldsymbol{\theta}, \boldsymbol{X}_{1:n})$ (Andrieu et al., 2010), such that if $\pi(\boldsymbol{\theta})$ is the prior and we apply an Metropolis-Hastings algorithm directly on $\boldsymbol{\theta}$ using $\hat{p}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n}) \propto \pi(\boldsymbol{\theta})\hat{\mathcal{L}}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$, it actually produces draws from the exact posterior $p(\boldsymbol{\theta}|\boldsymbol{X}_{1:n}) \propto \pi(\boldsymbol{\theta})\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$ (Andrieu et al., 2009).

4.3 The Synthetic Kalman Filter

Our so-called Synthetic Kalman Filter (SKF) attempts to combine features of both Kalman and particle filters. The name refers to the synthetic likelihoods approach of Wood (2010), where the intractable likelihood at $\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{X}_{\text{obs}})$ is approximated by a multivariate Gaussian density evaluated on a set of summary statistics $\boldsymbol{S}_{\text{obs}} = g(\boldsymbol{X}_{\text{obs}})$, with mean and variance estimated from a Monte Carlo sample from $p(\boldsymbol{S}|\boldsymbol{\theta})$. Similarly, our SKF approximation estimates $p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t})$ by a multivariate Gaussian, with mean and variance estimated by an importance sampler specifically tailored to the conditionally Gaussian structure of GG-SSMs. The details of the SKF procedure are presented in Algorithm 4.3.

In the first step, denote M as the number of the particles we generate from the filtering density. Unlike the particle filter that updating the particle conditions on previous particle, each $\boldsymbol{Z}_t^{(i)}$, $i = 1, \ldots, M$ is drawn from the same Gaussian distribution in the SKF. Step 2 prepares the weighted sample from $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t+1})$ by an importance sampler for step 3 and the weights in step 2 are due to that

$$p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t+1}) \propto p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_t, \boldsymbol{Z}_t)p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t}).$$

In step 3, note that $p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}, \boldsymbol{Z}_{t+1}|\boldsymbol{X}_t, \boldsymbol{Z}_t)$ follows $\mathcal{N}_{p+q}(\boldsymbol{\Lambda}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t), \boldsymbol{\Omega}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t))$, so the conditional

Algorithm 4.3: Synthetic Kalman Filter

Input : $\boldsymbol{\theta}$; $\boldsymbol{Z}_1|\boldsymbol{X}_1, \boldsymbol{\theta} \sim \mathcal{N}_q(\boldsymbol{\lambda}_1, \boldsymbol{\Psi}_1)$

Output: $\hat{\mathcal{L}}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$

1 Start the algorithm with $t = 1$ by drawing a Monte Carlo sample of $\boldsymbol{Z}_t^{(1)}, \ldots, \boldsymbol{Z}_t^{(M)} \overset{\text{iid}}{\sim} \mathcal{N}_q(\boldsymbol{\lambda}_t, \boldsymbol{\Psi}_t)$.

2 Calculate the weights of the sample $\omega_i = \frac{p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_t, \boldsymbol{Z}_t^{(i)})}{\sum_{j=1}^{M} p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_t, \boldsymbol{Z}_t^{(j)})}$.

3 The updated sufficient statistics are

$$\begin{aligned}
\boldsymbol{\lambda}_{t+1} &= \sum_{i=1}^{M} \omega_i f(\boldsymbol{Z}_t^{(i)}), \\
\boldsymbol{\Psi}_{t+1} &= \sum_{i=1}^{M} \omega_i \left[g(\boldsymbol{Z}_t^{(j)}) + (f(\boldsymbol{Z}_t^{(j)}) - \boldsymbol{\lambda}_{t+1})(f(\boldsymbol{Z}_t^{(j)}) - \boldsymbol{\lambda}_{t+1})' \right].
\end{aligned}$$

where functions f and g are conditional mean and variance of $p(\boldsymbol{Z}_{t+1}|\boldsymbol{Y}_t, \boldsymbol{\theta})$ that

$$\begin{aligned}
f(\boldsymbol{Z}_t) &= \boldsymbol{\Lambda}_t^{\mathbf{Z}} + \left(\boldsymbol{\Omega}_t^{\mathbf{XZ}}\right)' \left(\boldsymbol{\Omega}_t^{\mathbf{XX}}\right)^{-1} \left(\boldsymbol{X}_{t+1} - \boldsymbol{\Lambda}_t^{\mathbf{X}}\right), \\
g(\boldsymbol{Z}_t) &= \boldsymbol{\Omega}_t^{\mathbf{ZZ}} - \left(\boldsymbol{\Omega}_t^{\mathbf{XZ}}\right)' \left(\boldsymbol{\Omega}_t^{\mathbf{XX}}\right)^{-1} \left(\boldsymbol{\Omega}_t^{\mathbf{XZ}}\right),
\end{aligned} \tag{4.11}$$

where the statistics $\boldsymbol{\Lambda}_t^{\mathbf{X}}$, $\boldsymbol{\Lambda}_t^{\mathbf{Z}}$, $\boldsymbol{\Omega}_t^{\mathbf{X}}$, $\boldsymbol{\Omega}_t^{\mathbf{Z}}$ and $\boldsymbol{\Omega}_t^{\mathbf{XZ}}$ are from (4.9).

4 The mean and variance of the Gaussian approximation of the marginal transition density $p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_t)$ can be calculated as

$$\begin{aligned}
\hat{\boldsymbol{\Lambda}}_{t+1}^{\mathbf{X}} &= \frac{1}{n} \sum_{i=1}^{M} \boldsymbol{\Lambda}_t^{\mathbf{X}}(\boldsymbol{Z}_t^{(i)}) \\
\hat{\boldsymbol{\Omega}}_{t+1}^{\mathbf{XX}} &= \frac{1}{n} \sum_{i=1}^{M} \left(\boldsymbol{\Omega}_t^{\mathbf{XX}}(\boldsymbol{Z}_t^{(i)}) + (\boldsymbol{\Lambda}_t^{\mathbf{X}}(\boldsymbol{Z}_t^{(i)}) - \hat{\boldsymbol{\Lambda}}_{t+1}^{\mathbf{X}})(\boldsymbol{\Lambda}_t^{\mathbf{X}}(\boldsymbol{Z}_t^{(i)}) - \hat{\boldsymbol{\Lambda}}_{t+1}^{\mathbf{X}})' \right)
\end{aligned} \tag{4.12}$$

where $\Lambda_t^{\mathbf{X}}(\boldsymbol{Z}_t^{(i)})$ means we plug $\boldsymbol{Z}_t^{(i)}$ into the place of $\boldsymbol{Z}_t$ in $\boldsymbol{\Lambda}_t^{\mathbf{X}}$ and $\boldsymbol{\Omega}_t^{\mathbf{XX}}(\boldsymbol{Z}_t^{(i)})$ means we plug $\boldsymbol{Z}_t^{(i)}$ into the place of $\boldsymbol{Z}_t$ in $\boldsymbol{\Omega}_t^{\mathbf{XX}}$.

5 Given the updated filtering density $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}|\boldsymbol{X}_{1:t+1})$ in step 3, repeat step 1-4 by replacing t with $t + 1$.

6 Collect the Gaussian approximation of $p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t})$, $t = 1, \ldots, n-1$, so that $\hat{\mathcal{L}}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$ can be calculated by (4.3).

distribution can be derived as

$$\boldsymbol{Z}_{t+1}|\boldsymbol{X}_{t+1}, \boldsymbol{X}_t, \boldsymbol{Z}_t, \boldsymbol{\theta} \sim \mathcal{N}_q(f(\boldsymbol{Z}_t), g(\boldsymbol{Z}_t))$$

where functions $f(\boldsymbol{Z}_t)$ and $g(\boldsymbol{Z}_t)$ are defined in (4.11). The particle filter needs a sample from $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}|\boldsymbol{X}_{1:t+1})$, which is not required in the SKF. Instead, the SKF only needs the mean and variance of $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}|\boldsymbol{X}_{1:t+1})$, which can be evaluated by the laws of total expectation and variance of

$$\begin{aligned}
E\big[\boldsymbol{Z}_{t+1} \mid \boldsymbol{X}_{1:t+1}, \boldsymbol{\theta}\big] &= E\Big[E\big(\boldsymbol{Z}_{t+1}|\boldsymbol{X}_{1:t+1}, \boldsymbol{Z}_t, \boldsymbol{\theta}\big) \mid \boldsymbol{X}_{1:t+1}, \boldsymbol{\theta}\Big] = E\big[f(\boldsymbol{Z}_t)|\boldsymbol{X}_{1:t+1}, \boldsymbol{\theta}\big], \\
V\big[\boldsymbol{Z}_{t+1} \mid \boldsymbol{X}_{1:t+1}, \boldsymbol{\theta}\big] &= E\big[g(\boldsymbol{Z}_t) \mid \boldsymbol{X}_{1:t+1}, \boldsymbol{\theta}\big] + V\big[f(\boldsymbol{Z}_t) \mid \boldsymbol{X}_{1:t+1}, \boldsymbol{\theta}\big],
\end{aligned}$$

over a sample from $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t+1})$. Essentially, this step solves the update formula (4.7) by an importance sampler, while the Kalman filter bypasses it by analytic results.

In step 4, the SKF method offers a straightforward way to approximate the model likelihood. The first two moments of the transition density of the observed process $p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t})$ are estimated by applying Monte Carlo (MC) method on the integral of (4.6) over the simulations from $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t})$. Hence, the sample from the first step can be reused in this step. Indeed, according to the integral (4.6), the estimated mean and variance (4.12) are from the laws of total expectation and variance of

$$\begin{aligned}
E\big[\boldsymbol{X}_{t+1} \mid \boldsymbol{X}_{1:t}, \boldsymbol{\theta}\big] &= E\Big[E\big(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t}, \boldsymbol{Z}_t, \boldsymbol{\theta}\big) \mid \boldsymbol{X}_{1:t}, \boldsymbol{\theta}\Big] = E\big[\boldsymbol{\Lambda}_t^{\mathbf{X}}|\boldsymbol{X}_{1:t+1}, \boldsymbol{\theta}\big], \\
V\big[\boldsymbol{X}_{t+1} \mid \boldsymbol{X}_{1:t}, \boldsymbol{\theta}\big] &= E\big[\boldsymbol{\Omega}_t^{\mathbf{X}} \mid \boldsymbol{X}_{1:t}, \boldsymbol{\theta}\big] + V\big[\boldsymbol{\Lambda}_t^{\mathbf{X}} \mid \boldsymbol{X}_{1:t}, \boldsymbol{\theta}\big].
\end{aligned}$$

Due to that we approximate $p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t})$ as Gaussian, the first two moments are sufficient.

Given the detailed algorithms of the Kalman filter, the particle filter and the SKF, we can summarize some important differences among those filtering methods. The applicable Gaussian models for the Kalman-type variants, especially for the extended Kalman filter, are usually constrained to apply to the conventional state-space model. To our knowledge, Kalman-type variants have rarely been applied to a model with non-zero $\boldsymbol{\Omega}_t^{\mathbf{XZ}}$ (4.9) and/or $\boldsymbol{\Omega}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t)$ (4.2) involving model variables. The particle filter is applicable to any models

	Filtering Density	Monte Carlo	Importance sampler	Robustness
Kalman-type variants	approximate Gaussian	Possible	No, usually determined weights	High
SKF	approximate Gaussian	Yes	Yes; exact given the Gaussian assumption	No particle degeneration
Particle Filter	Weight particles (exact)	Yes	Yes; exact	Particle degeneration

Table 4.1: Comparison of filtering methods for nonlinear models.

which fall inside the category of GG-SSMs (4.2). Our proposed SKF method requires the Gaussian assumptions to be close to their true densities. By our experience with the SV models, we often can find a transformation of the model variables such that the densities on the transformed variables fit closely to the Gaussian assumptions. This transformation can be easily applied to the SDE model (4.1) by Itō's lemma before being discretized into the GG-SSM (4.2). Given the GG-SSM (4.2) is a Gaussian model and has an SDE model origin, we should have a good chance of finding such an desirable transformation for an arbitrary GG-SSM. In fact, we will show this transformation by the exponential OU model of Scott (1987); Fouque et al. (2000) later in Section 4.6. In Table 4.1, we summarize some other important points. The SKF model shares the Gaussian assumptions with the Kalman-type variants. There are some special Kalman-type variants, such as the ensemble Kalman filter, which do employ the MC method. They are often designed to have equal weights instead of the importance sampler as in the SKF and the particle filter. The importance sampler in the SKF method does not have the degeneration problem of the particle filter, since the weighted particle information is summarized into the filtering mean and variance of the latent variable at the end of each update.

4.4 Efficient Bayesian Inference

In Step 1 of the SKF Algorithm 4.3, denote

$$\boldsymbol{Z}_t^{(i)} = \boldsymbol{\lambda}_t + \boldsymbol{\Psi}_t^{1/2}\zeta_t^{(i)}, \tag{4.13}$$

where $\zeta_t^{(i)}$ is the noise variable/vector and $\boldsymbol{\Psi}_t^{1/2}$ is the lower Cholesky factor of $\boldsymbol{\Psi}_t$. The Monte Carlo variance of the SKF likelihood estimator $p_{\text{SKF}}(\boldsymbol{X}_{1:n}|\boldsymbol{\theta})$ can be greatly reduced by using the same set of noise variables $\boldsymbol{\zeta}_t = \left[\zeta_t^{(i)}\right]_{i=1,\ldots,M}$ for every value of $\boldsymbol{\theta}$. This has two important consequences for speeding up inference. First, the posterior density can be approximated by its asymptotic Gaussian density, which requires significantly less evaluations of the likelihood than an MCMC algorithm requires. The particle filter usually does not share this property since the randomized resampling step is necessary to prevent particle degeneracy, which adds inherent randomness into its likelihood evaluation and makes it almost impossible to find the asymptotic density by an optimizer. Note that the model likelihood by the continuous particle filter (Malik and Pitt, 2011; Pitt et al., 2014) is a function of the parameters but the applications are limited to models with univariate $\boldsymbol{X}_t$. Second, the Quasi-Monte Carlo can be easily applied.

4.4.1 Mode-Quadrature Posterior Approximation

Note that the SKF method offers the approximation of the observed data likelihood $p_{\text{SKF}}(\boldsymbol{X}_{1:n}|\boldsymbol{\theta})$ as one direct function on the parameters. The approximate Bayesian inference can be conducted on just the parameter space since

$$p(\boldsymbol{\theta}|\boldsymbol{X}_{1:n}) = \frac{\pi(\boldsymbol{\theta})p(\boldsymbol{X}_{1:n}|\boldsymbol{\theta})}{p(\boldsymbol{X}_{1:n})},$$

where $\pi(\boldsymbol{\theta})$ is the prior and the approximate posterior $p_{\text{SKF}}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$ can be evaluated by $\pi(\boldsymbol{\theta})p_{\text{SKF}}(\boldsymbol{X}_{1:n}|\boldsymbol{\theta})$. The dimension of the parameter space is much lower than the dimension of the augmented space for some MCMC algorithms, e.g., the one described in Appendix B. The reduced dimension usually makes the posterior sample much more efficient. However, the cost per likelihood evaluation is typically high since the latent variables have been

integrated out during the likelihood evaluation. Hence, a scheme to draw from the posterior density that needs fewer likelihood evaluations is desirable. For this reason, we use an optimizer to search for the maximum a posteriori probability estimate (MAP) of $\boldsymbol{\theta}_{MAP} = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$, which is the mode of the posterior density, and then construct the approximate posterior distribution as a multivariate Gaussian distribution of

$$\mathcal{N}\left(\boldsymbol{\theta}_{MAP}, [-\boldsymbol{H}(\boldsymbol{\theta}_{MAP})]^{-1}\right), \tag{4.14}$$

where $\boldsymbol{H}(\boldsymbol{\theta}_{MAP})$ is the Hessian matrix of the log-posterior density evaluated at $\boldsymbol{\theta}_{MAP}$ by a numerical method since the analytic Hessian matrix is not available. Indeed, the posterior density we operate on is $p_{\text{SKF}}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$ when we use the SKF method. We refer to this posterior approximation scheme as the Mode-Quadrature (MQ) method.

According to Hartigan (1983), the posterior distribution is asymptotically Gaussian, with the maximum likelihood estimate (MLE) as the mean and the inverse of the Fisher information matrix as the variance-covariance matrix. Under the non-informative prior that $\pi(\boldsymbol{\theta}) \propto 1$, the MAP is essentially the same as the MLE since the prior is constant on the parameter space. In addition, the negative Hessian matrix $-\boldsymbol{H}(\boldsymbol{\theta}_{MAP})$ is equivalent to the observed Fisher information. Hence, the Gaussian distribution (4.14) is in fact the asymptotic posterior distribution.

Reparameterizing is very helpful to the numerical stability of the optimizer and the numerical Hessian solver, and further improves the quality of the estimation. In practice, we reparameterize $\boldsymbol{\theta}$ by its unbounded version $\tilde{\boldsymbol{\theta}}$, and apply the non-informative prior on the unbounded version such that $\pi(\tilde{\boldsymbol{\theta}}) \propto 1$. For example, if one parameter θ is defined to be positive, then $\tilde{\theta} = \log(\theta)$ and $\pi(\tilde{\theta}) \propto 1 \Leftrightarrow \pi(\theta) \propto \frac{1}{\theta}$; if one parameter θ is defined to be between -1 and 1, then $\tilde{\theta} = \log(\frac{\theta+1}{1-\theta})$ and $\pi(\tilde{\theta}) \propto 1 \Leftrightarrow \pi(\theta) \propto \frac{2}{1-\theta^2}$. In fact, we only encounter these two reparametrize scenarios in the models of interest later in Section 4.5.

Now, we can write down Algorithm 4.4 of generating a sample from the approximate posterior density by the Mode-Quadrature method with reparametrization.

Algorithm 4.4: Mode-Quadrature Posterior Approximation

1 Reparameterize $\boldsymbol{\theta}$ by its unbounded version $\tilde{\boldsymbol{\theta}} = \boldsymbol{g}(\boldsymbol{\theta})$, where $\boldsymbol{g}$ is the vector of functions that covert $\boldsymbol{\theta}$ element-wise.

2 Set the non-informative prior on $\tilde{\boldsymbol{\theta}}$ such that $\pi(\tilde{\boldsymbol{\theta}}) \propto 1$.

3 Find $\tilde{\boldsymbol{\theta}}_{MAP}, \boldsymbol{H}(\tilde{\boldsymbol{\theta}}_{MAP})$, by the optimizer and the numeric Hessian matrix solver on the log-posterior density $\log p(\tilde{\boldsymbol{\theta}}|\boldsymbol{X}_{1:n})$.

4 Sample $\tilde{\boldsymbol{\theta}}$ from $\mathcal{N}\left(\tilde{\boldsymbol{\theta}}_{MAP}, \left[-\boldsymbol{H}(\tilde{\boldsymbol{\theta}}_{MAP})\right]^{-1}\right)$.

5 Covert the $\tilde{\boldsymbol{\theta}}$ sample by the inverse function of $\boldsymbol{g}(\cdot)$, such that a posterior sample can be obtained by $\boldsymbol{\theta} = \boldsymbol{g}^{-1}(\tilde{\boldsymbol{\theta}})$.

4.4.2 Variance Reduction by Quasi-Monte Carlo

It is well-known that the efficiency of the Quasi-Monte Carlo estimator via low-discrepancy sequence is significantly higher than the ordinary Monte Carlo estimator. In this subsection, we want to show the SKF method is compatible to the Quasi-Monte Carlo (QMC) implementation.

In the first step of the SKF Algorithm 4.3, we need to simulate from a Gaussian filtering density. Unlike the particle filter that each simulation is based on one specific particle of previous state, all of the simulations in this step are from the same Gaussian distribution. Note that the objective is to find the MC estimators based on those simulations, the low-discrepancy sequence of the QMC can be used in this step. Specifically, we replace the Gaussian noise variables $\boldsymbol{\zeta}_t$ of (4.13) by the Gaussian deviated q-dimensional low-discrepancy sequences.

Note that the Gaussian noise $\boldsymbol{\zeta}_t$ introduce extra randomness into the evaluation of the posterior density. If each posterior density evaluation of the SKF uses a different Gaussian sample, the optimizer on $\boldsymbol{\theta}$ will be searching an ever-changing surface since $\boldsymbol{\zeta}_t$ contribute variation to the posterior density. Hence, we need to use the same set of low-discrepancy

points for every posterior density evaluation in Algorithm 4.4, so that the optimizer can search a fixed surface depending only on the parameters.

Note that the number of particles M required for the SKF method should be less than for the particle filter since we only need to estimate the first two moments. With the help of the QMC, we can further reduce the number of particles of the SKF method. Since the computational cost of evaluating the model likelihood by the SKF method grows linearly with the number of particles, the implementation of the QMC in the SKF posterior evaluation provides significant reduction in the computational time.

4.5 Numerical Evidence

In this section, we consider the discretized exponential OU model (Scott, 1987; Fouque et al., 2000) that has a form

$$\begin{aligned} \log S_{t+1} &= \log S_t + (\alpha - \tfrac{1}{2}V_t^2)\Delta\mathrm{T} + V_t\epsilon_t^S, \\ \log V_{t+1} &= \log V_t + \gamma(\mu - \log V_t)\Delta\mathrm{T} + \sigma\epsilon_t^V, \end{aligned} \qquad \begin{pmatrix} \epsilon_t^S \\ \epsilon_t^V \end{pmatrix} \overset{\text{iid}}{\sim} \mathcal{N}_2\left(\mathbf{0}, \Delta\mathrm{T}\begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right), \tag{4.15}$$

where $\gamma > 0$, $\sigma > 0$ and $-1 < \rho < 1$. This model (4.15) can be cast into the GG-SSM (4.2) by setting $\boldsymbol{X}_t = \log S_t$ and $\boldsymbol{Z}_t = \log V_t$. The most important reason for using $(\log S_t, \log V_t)$ is that the filtering density $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t-1})$ and the marginal transition density $p_{\boldsymbol{\theta}}(\boldsymbol{X}_t|\boldsymbol{X}_{1:t-1})$ are extremely close to Gaussian distributions. One other reason is that (S_t, V_t) must be positive but $(\log S_t, \log V_t)$ are unbounded. If we use (S_t, V_t), the QMC simulation of $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t})$ in the first step of Algorithm 4.3 can fall outside the support of V_t, since V_t must be positive.

The first dataset we investigate in this section is a simulated dataset of 1000 daily observations ($\Delta\mathrm{T} = \frac{1}{252}$) from the single-asset eOU model (4.15) under the set of parameters as shown in Table 4.2. This set of parameters is chosen according to our experience of fitting the SPX index. The volatility V_t is considered as latent in the analysis on this dataset.

We are also interested in the multivariate SV model proposed in the last chapter that

α	γ	μ	σ	ρ
0.1	5.0	-2.0	1.0	-0.6

Table 4.2: True parameters of the single-asset eOU model.

has a form

$$\begin{aligned} \log S_{i,t+1} &= \log S_{i,t} + (\alpha_i - \tfrac{1}{2}V_{i,t}^2)\Delta\mathrm{T} + V_{i,t}\epsilon_{i,t}^S, \\ \log V_{i,t+1} &= \log V_{i,t} + \gamma_i(\mu_i - \log V_{i,t})\Delta\mathrm{T} + \sigma_i\epsilon_{i,t}^V, \\ \log V_{0,t+1} &= \log V_{0,t} + \gamma_0(\mu_0 - \log V_{0,t})\Delta\mathrm{T} + \sigma_0\epsilon_{0,t}^V, \end{aligned} \qquad \begin{pmatrix} \boldsymbol{\epsilon}_t^S \\ \boldsymbol{\epsilon}_t^V \\ \epsilon_{0,t}^V \end{pmatrix} \overset{\text{iid}}{\sim} \mathcal{N}_{2q+1}\left(\mathbf{0}, \Delta\mathrm{T}\boldsymbol{R}\right), \quad (4.16)$$

where $\gamma_i > 0$, $\sigma_i > 0$, $i = 1, \ldots, q$, $\boldsymbol{\epsilon}_t^S = (\epsilon_{1,t}^S, \ldots, \epsilon_{q,t}^S)$, $\boldsymbol{\epsilon}_t^V = (\epsilon_{1,t}^V, \ldots, \epsilon_{q,t}^V)$ and $\boldsymbol{R}$ is the correlation matrix with entries defined as

$$\begin{aligned} &\mathrm{cor}(\epsilon_{i,t}^S, \epsilon_{i,t}^V) = \rho_i, \qquad \mathrm{cor}(\epsilon_{i,t}^V, \epsilon_{j,t}^V) = \tau_i\tau_j, \qquad \mathrm{cor}(\epsilon_{i,t}^S, \epsilon_{j,t}^V) = \rho_i\tau_i\tau_j, \\ &\mathrm{cor}(\epsilon_{i,t}^S, \epsilon_{j,t}^S) = \rho_i\rho_j\tau_i\tau_j + \sqrt{(1-\rho_i^2)(1-\rho_j^2)}\,\omega_i\omega_j, \\ &\mathrm{cor}(\epsilon_{i,t}^S, \epsilon_{0,t}^V) = \tau_i\rho_i, \qquad \mathrm{cor}(\epsilon_{i,t}^V, \epsilon_{0,t}^V) = \tau_i, \qquad 1 \le i \ne j \le q. \end{aligned} \quad (4.17)$$

where $-1 < \tau_i < 1$, $-1 < \rho_i < 1$, $-1 < \omega_i < 1$. The same as the single-asset model, we cast this model (4.16) into the GG-SSM (4.2) with variables of $\boldsymbol{X}_t = (\log S_{1,t}, \ldots, \log S_{q,t}, \log V_{0,t})$ and $\boldsymbol{Z}_t = (\log V_{1,t}, \ldots, \log V_{q,t})$. The notation q refers to the asset number in this model (4.16) and also the number of the latent volatilities, which is compatible with its meaning of the dimension of the latent variables in (4.9). The observed variables in this model are the asset prices and the volatility factor, hence, $p = q + 1$.

In this section, we focus on the 2-asset version of this model (4.16) as an example of the multi-asset SV models. The second dataset we investigate is a simulated dataset of 1,000 daily observations ($\Delta\mathrm{T} = \frac{1}{252}$) from the 2-asset model (4.16) using the set of parameters as shown in Table 4.3. This set of parameters is chosen according to our experience with fitting financial stocks of JP Morgan and Citigroup. As mentioned, we set the volatility factor V_0 as observed and the volatilities V_1 and V_2 as latent. There is only one ω since $q = 2$ is a special case for the model (4.16) as we explained in Chapter 3 that ω_1 and ω_2 are non-distinguishable, hence, we combine they together by $\omega = \omega_1\omega_2$.

α_1	α_2	γ_1	γ_2	μ_1	μ_2	σ_1	σ_2	ρ_1	ρ_2	τ_1	τ_2	ω	γ_0	μ_0	σ_0
0.1	0.05	7	8	-1	-1.25	0.9	1.1	-0.6	-0.7	0.8	0.9	0.36	5	2.5	1.2

Table 4.3: True parameters of the 2-asset eOU model.

4.5.1 Accuracy of SKF for Volatility Filtering

In this subsection, we will check the SKF Gaussian approximation of the filtering density $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t})$. The analysis is performed on the two simulated dataset from the two eOU models, i.e., the single-asset eOU model (4.15) and the 2-asset eOU model (4.16), as described at the beginning of this Section 4.4. The volatility filtering for the two datasets conditions on the corresponding true parameters in Table 4.2 or Table 4.3.

The methods we used for the study in this subsection are as follows.

1. The AugMCMC method: this method refers to the MCMC algorithm (Lysy and Tong, 2017) on the augmented space of the latent variables and the parameters, which is detailed in Appendix B. Note that this method only offers the smoothing density instead of the filtering density. If we are interested in the filtering density $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t})$, we have to run this AugMCMC method on the dataset $\boldsymbol{X}_{1:t}$. In some sense, this is one drawback of this AugMCMC method, that it is not an on-line method as the filtering methods. The filtering density from this method is constructed by an MCMC sample of size 100,000. This is the first benchmark method since it draws from the exact posterior densities.

2. The SKF method: this is our proposed method. As mentioned, we use the low-discrepancy sequence to reduce the number of particles M. In particular, we use $M = 100$ Sobol low-discrepancy points for each update of both datasets. Since the filtering mean and variance are the only statistics obtained by the SKF method, the filtering sample is constructed by a simulated sample of size 100,000 from the approximate Gaussian filtering distribution. We expect this approximate Gaussian to be very close to the densities from the exact methods.

3. The particle filter (PF) method: the SIR implementation as stated in Algorithm 4.2.

We use $M = 1,000$ particles for the dataset from the single-asset model, as suggested by Andrieu et al. (2010) where they use $M = 1,000$ particles for a slightly more complicated SV model. For the dataset from the 2-asset model, we increase the number of particles to $M = 5,000$. To construct a sample size of 100,000, we resample from the 1,000 (or 5,000) particles according to their weights. This serves as the second benchmark method.

Figure 4.1 shows the comparison of the filtering densities based on the dataset from the single-asset model on ten randomly selected time points. As discussed before, we implement the model on the variables of $(\log S_t, \log V_t)$, so that the Gaussian approximation of the SKF method is actually on $\log V_t$. Hence, the densities of the volatilities on their regular scales are skewed as shown in Figure 4.1. It is reasonable to say that all of the three methods output extremely similar filtering densities. The densities of the particle filter are relatively coarse since they are based on a resampling from 1,000 particles. However, the weighted mean and variance from the 1,000 particles are the same as those from the AugMCMC algorithm.

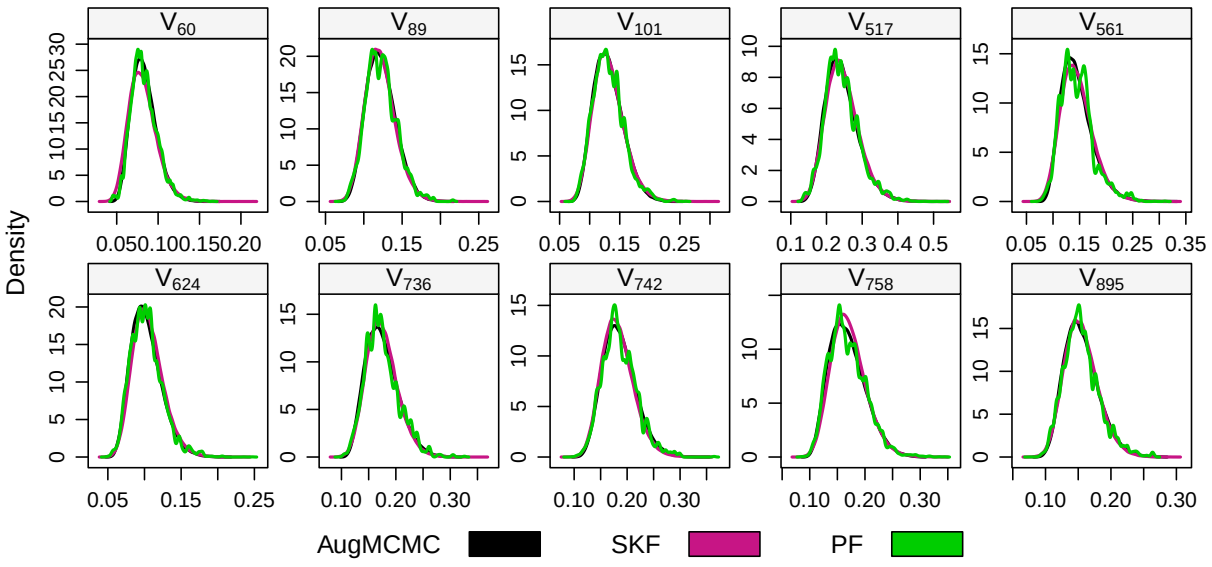


Figure 4.1: Volatility filtering density comparison of the single-asset model.

Figure 4.2 shows the comparison of the filtering densities based on the dataset from the 2-asset model on four randomly selected time points. Note that there are two latent volatilities in the 2-asset model, hence, we demonstrate the 2-dimensional densities by contour plot. It appears that the SKF approximate densities are not as close to the AugMCMC benchmark densities as the single-asset eOU model result (Figure 4.1). Actually, the contour plots seem to amplify the differences between the compared densities visually. Figure 4.3 shows the marginal filtering densities of the same time points as Figure 4.2. On the point of $V_{1,918}$, the SKF estimated mean is slightly biased. The other densities are still very close to the benchmark densities. By comparing all of the posterior means between the AugMCMC and SKF, we find this happens rarely. Those results show that the Gaussian approximation of the 2-dimensional filtering density is reasonable. In fact, the contour plots of the particle filter confirms the importance of the Gaussian approximation since we cannot obtain a reasonably smooth density by particle filter with 5,000 particles. We can imagine that when the dimension of the latent variables grows, the particle filter will need more and more particles which is a computational burden that we cannot afford. We want to explain another point that the volatilities are not very correlated as it is shown in the contour plots. Since the correlation matrix (4.17) of the multi-asset model (4.16) builds the correlation between the random increments of the volatilities only through the volatility factor, which is observed in this case, there should be no correlation between the volatilities theoretically.

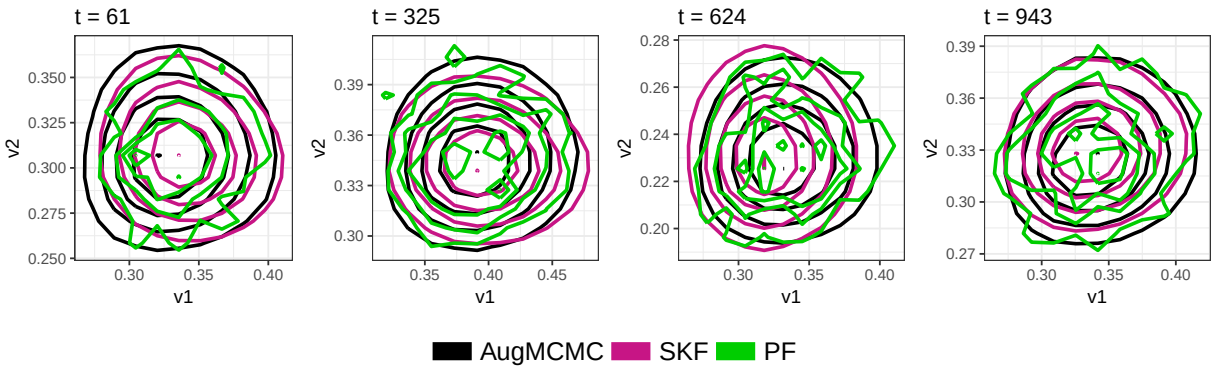


Figure 4.2: Volatility filtering density comparison of the 2-asset model.

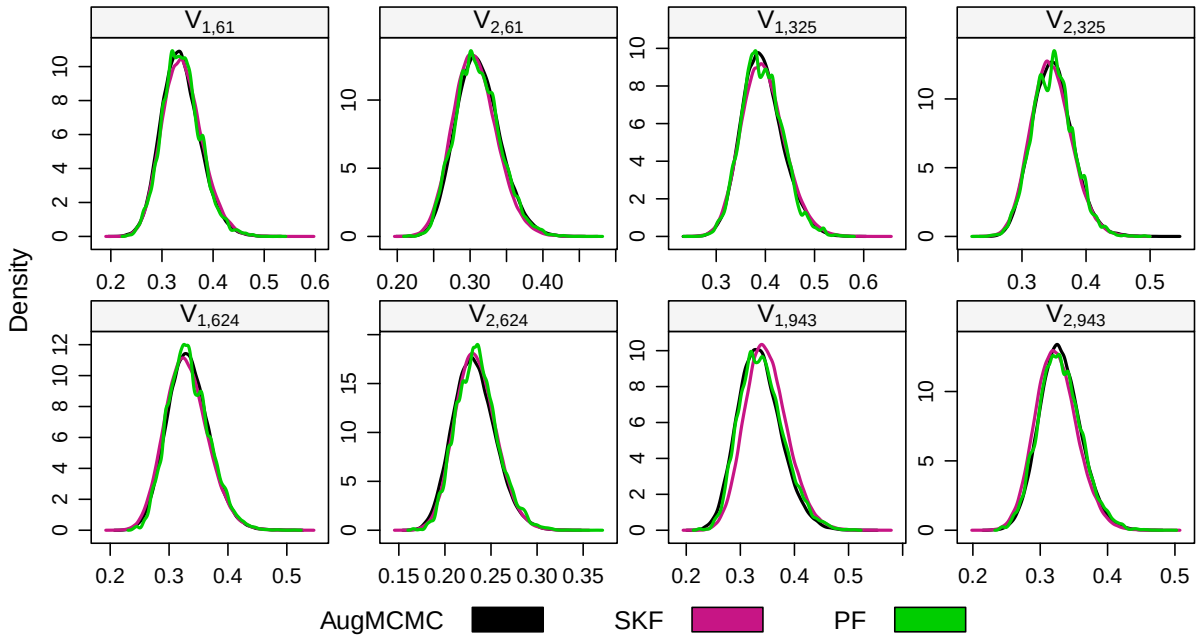


Figure 4.3: Marginal volatility filtering density comparison of the 2-asset model.

To summarize, the evidence we present for the filtering densities of the two eOU models show the Gaussian approximation of the filtering density in the proposed SKF method is very close to the true density provided by the AugMCMC method. The 2-dimensional demonstration further confirms that a weighted particle representation of a high-dimensional problem requires a large number of particles using the particle filter.

4.5.2 Accuracy of SKF for Parameter Estimation

In this subsection, we further assess the Gaussian assumptions in the SKF method. The parameter estimation is often the goal of many studies instead of obtaining the filtering density. Thus, we skip the step of checking the Gaussian approximation on $p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t})$ and directly focus on the approximate posterior density of the parameters $p_{\text{SKF}}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n})$, which in some degree is the summary of all of the Gaussian approximation on $p_{\boldsymbol{\theta}}(\boldsymbol{X}_{t+1}|\boldsymbol{X}_{1:t})$ and $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:t})$, $t = 1, \ldots, n-1$.

We compare the following two inference methods on the two simulated datasets.

1. The AugMCMC method: the details can be found in Appendix B. We draw a sample of size 1 million using this AugMCMC method.

2. The SKF method: we use the MQ Algorithm 4.4 with $M = 100$ low-discrepancy points.

Figure 4.4 shows the posterior density comparison of the 1-asset dataset. The dashed lines mark the true parameter values as in Table 4.2. We notice that there are small differences between the two curves. However, the differences in the modes of the two densities are almost negligible. We also check the variance-covariance matrix of the posterior sample from the AugMCMC method and it is very close to the asymptotic variance from the MQ method using SKF likelihood. Here we skip the details of the two matrices. In other words, the posterior density approximated by our SKF method is very close to the true posterior for the single-asset eOU model (4.15).

In terms of the computational time comparison, we will discuss it later in Section 4.6.4. In brief, we can obtain the above results around 100 times faster using the SKF with the MQ method 4.4 than the AugMCMC method.

Figure 4.5 shows the posterior density comparison of the 2-asset dataset. The dashed lines mark the true parameter values as in Table 4.3. Again, the two sets of the marginal posterior densities are remarkably similar and the variance-covariance matrices are almost the same, which are not shown here. We find some posterior modes are quite far away from their true values, e.g., ρ_1, γ_0, however, both methods agree on the posterior densities of those parameters. Therefore it might be that the true parameters are not reflected properly by this specific dataset. The computational time of both methods grow significantly as the dimension of the model grows. It takes roughly 5 hours to finish the calculation for 1 million draws from the AugMCMC method and around 250 seconds for the MQ method with posterior density estimated by the SKF method. The computational gain is significant, but the loss on the approximation is miniscule.

In summary, by analyzing both datasets of the single-asset model and the 2-asset model, the approximate posterior densities by the SKF are extremely close to the benchmark

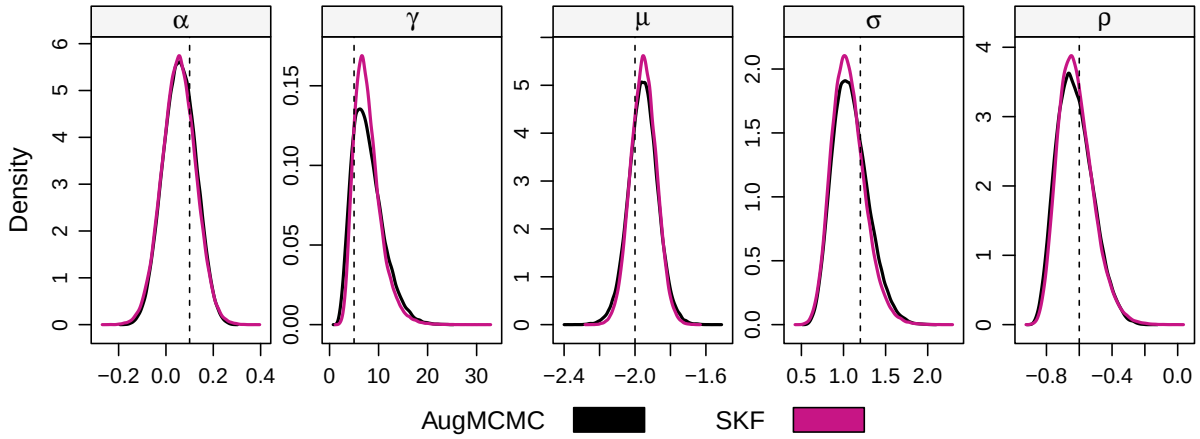


Figure 4.4: Parameter posterior density comparison of the single-asset model.

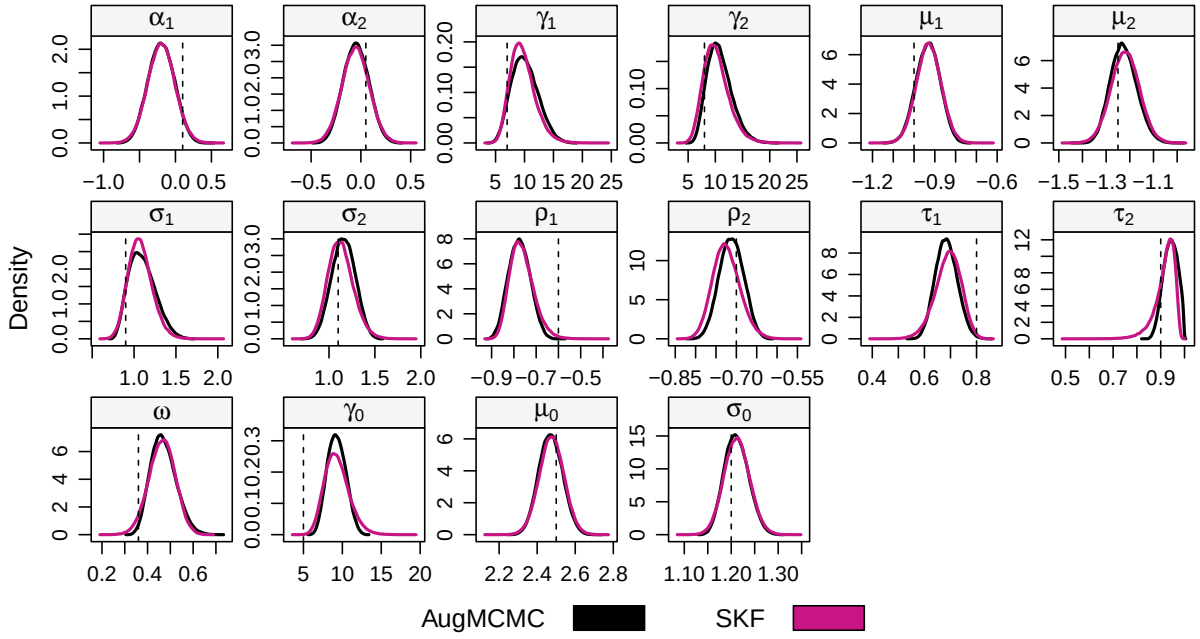


Figure 4.5: Parameter posterior density comparison of the 2-asset model.

densities from the AugMCMC method. We will provide more comparisons on the posterior densities in the later simulation study.

4.5.3 Number of Particles

In the previous subsections, we use $M = 100$ particles in the SKF method for both the 1-asset model and the 2-asset model. It is very important for us to decide the number of the particles. On one hand, the computational difficulty of the SKF method is in a linear relationship with the number of the particles. On the other hand, a smaller number of particles means the approximations of the Gaussian mean and variance might be in poor states. In summary, it is important to find an appropriate size of the particles.

In this subsection, we compare three levels of number of particles of $M = 30$, 100 and 300 on both the single-asset eOU model (4.15) and 2-asset eOU model (4.16) by the filtering density and the posterior density. To have a more comprehensive perspective, we expand the simulated datasets to $N_s = 20$ for both of the two models using the corresponding sets of parameters in Table 4.2 and Table 4.3. Each of the path contains $N = 1000$ daily observations.

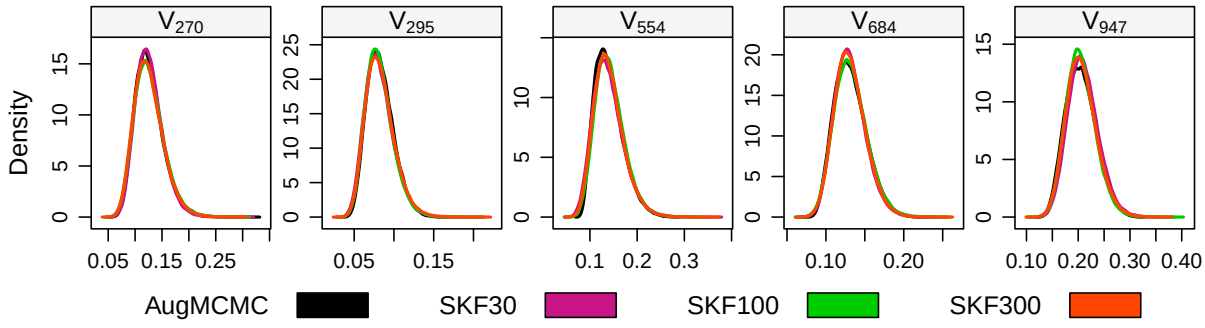


Figure 4.6: Volatility filtering density comparison using different number of particles of 1-asset model.

First of all, we check the differences of the filtering means and variances. In Figure 4.6 we show some randomly selected examples of the filtering densities of the single-asset

dataset and in Figure 4.7 we show some randomly selected examples of the filtering densities of the 2-asset dataset. The AugMCMC method serves as the benchmark here. It is very hard to tell the differences among the densities in Figure 4.6. In Figure 4.7, we are able to pick up some differences, but again the densities are not as obvious if viewed marginally. Hence, we need more summary statistics on the filtering means and variances of all of the $N_s = 20$ paths.

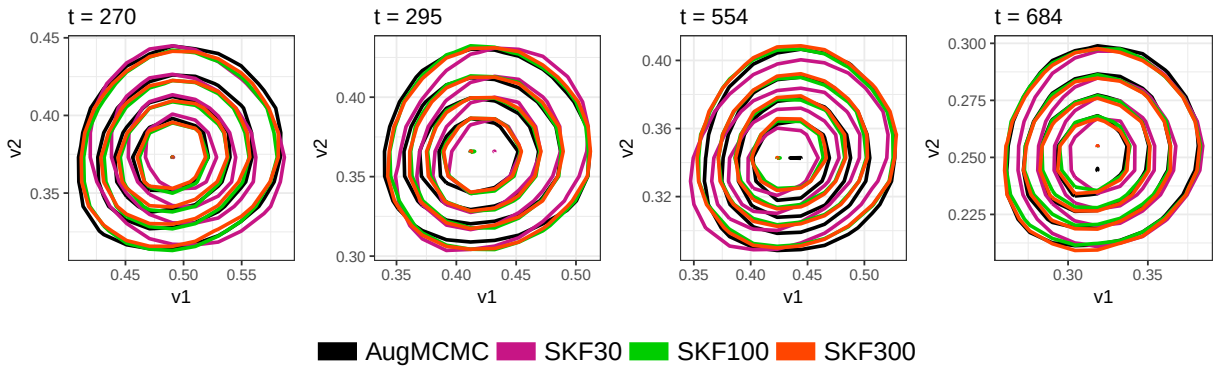


Figure 4.7: Volatility filtering density comparison using different number of particles of 2-asset model.

Note that in each of the $N_s = 20$ paths, there are 1000 latent volatilities for the single-asset eOU model and 1000×2 latent volatilities for the 2-asset model. Denote $\mathrm{E}(\boldsymbol{V}_{i,t,j})$ and $\mathrm{sd}(\boldsymbol{V}_{i,t,j})$ as the mean and standard error of the AugMCMC posterior sample of $\boldsymbol{V}_{i,t}$ in the jth dataset, and also denote $\mathrm{E}_{\mathrm{SKF}}(\boldsymbol{V}_{i,t,j})$ and $\mathrm{sd}_{\mathrm{SKF}}(\boldsymbol{V}_{i,t,j})$ as those by the SKF method. Table 4.4 shows the average of the absolute differences

$$\frac{1}{20Nq}\sum_{i,t,j}|\mathrm{E}(\boldsymbol{V}_{i,t,j}) - \mathrm{E}_{\mathrm{SKF}}(\boldsymbol{V}_{i,t,j})| \quad \text{and} \quad \frac{1}{20Nq}\sum_{i,t,j}|\mathrm{sd}(\boldsymbol{V}_{i,t,j}) - \mathrm{sd}_{\mathrm{SKF}}(\boldsymbol{V}_{i,t,j})|,$$

and the average of the relative differences

$$\frac{1}{20Nq}\sum_{i,t,j}\frac{|\mathrm{E}(\boldsymbol{V}_{i,t,j}) - \mathrm{E}_{\mathrm{SKF}}(\boldsymbol{V}_{i,t,j})|}{\mathrm{E}(\boldsymbol{V}_{i,t,j})} \quad \text{and} \quad \frac{1}{20Nq}\sum_{i,t,j}\frac{|\mathrm{sd}(\boldsymbol{V}_{i,t,j}) - \mathrm{sd}_{\mathrm{SKF}}(\boldsymbol{V}_{i,t,j})|}{\mathrm{sd}(\boldsymbol{V}_{i,t,j})}.$$

The results meet our expectation that as the number of particles increases, the average absolute difference decreases. The gain of increasing the number of particles from $M = 100$

to $M = 300$ is not as large as the gain of increasing the number of particles from $M = 30$ to $M = 100$. The mean estimation is much more accurate than standard deviation estimation according to the relative difference results. We further increase the number of particles and find that the around 0.4% bias of the 1-asset model and around 0.9% bias of the 2-asset model on the mean estimation are very hard to eliminate by using even larger number of particles in the SKF. We suspect that a small amount of the filtering densities on $\log V_t$ are analytically skewed which systematically cannot be picked up by the SKF Gaussian approximations. For example, the plot of $V_{1,943}$ in Figure 4.3 shows the log-Gaussian density of the SKF seems to match better with the analytic densities on the tails than on the mode, which suggests the true filtering density on $\log V_{1,943}$ might not be so symmetric and cannot be approximated very well by a Gaussian density.

		Absolute Difference			Relative Difference		
		30	100	300	30	100	300
1-asset	Mean	0.023	0.012	0.009	1.12%	0.59%	0.44%
	s.d.	0.014	0.007	0.005	7.15%	3.77%	2.63%
2-asset	Mean	0.015	0.011	0.010	1.39%	1.03%	0.98%
	s.d.	0.007	0.003	0.002	6.63%	2.70%	1.82%

Table 4.4: Average absolute differences of the filtering Mean and s.d. of the volatilities.

Generally speaking, the larger the number of particles leads to more accurate volatility filtering densities by the results of Table 4.4. Now we move on to check whether this conclusion also applies to the estimation of the posterior densities.

Figure 4.8 shows the posterior marginal densities of two randomly selected datasets from the 20 single-asset simulated datasets. We spot small differences between the posterior densities generated by different sizes of the particles, however, they are not significantly. The three sets of the SKF densities seem to cluster but are a little deviated from the benchmark AugMCMC density, especially for γ. This suggests that the SKF method together with the MQ algorithm 4.4 produces an approximate posterior density which is slightly biased at least for some parameters. There is no significant evidence to show that the posterior densities of the 300-particle scheme is closer to the benchmark densities than

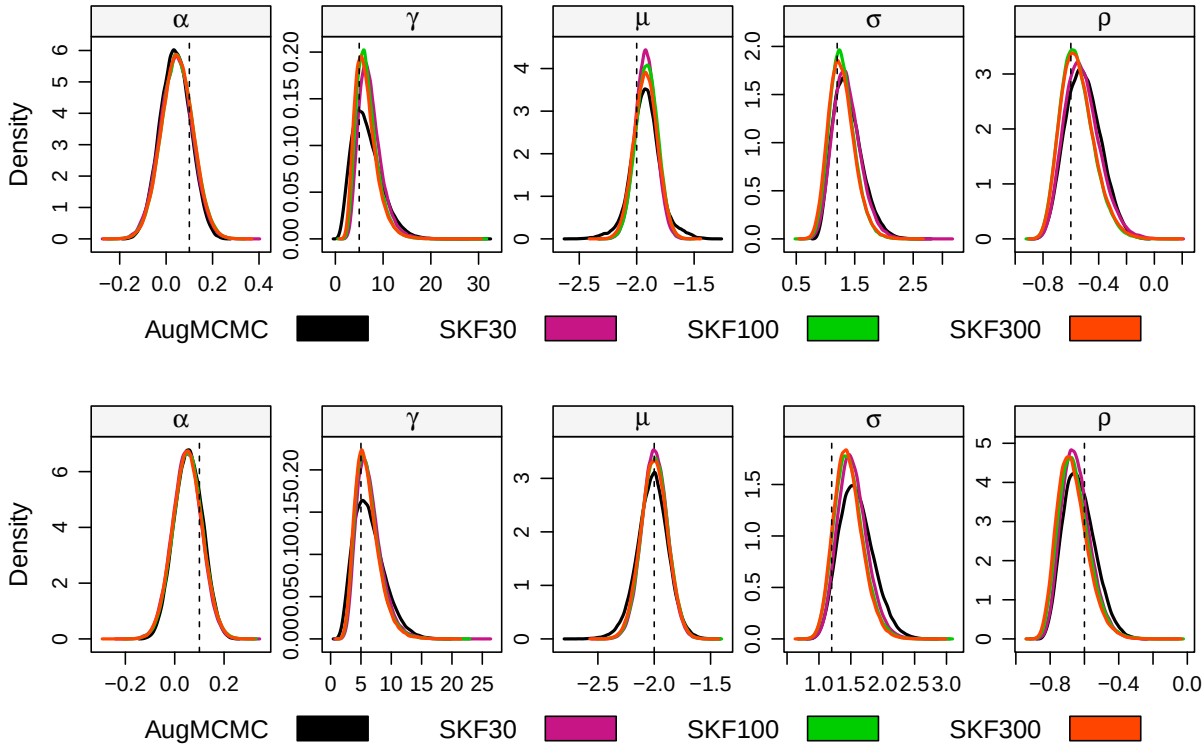


Figure 4.8: Posterior density comparison using different number of particles on two randomly selected datasets of 1-asset model. *Top:* the first dataset; *bottom:* the second dataset.

those of the 30-particle scheme. This result is confirmed by summarizing the posterior results of the 20 paths. Denote the marginal posterior mean and standard deviation of $\boldsymbol{\theta}$ as $\mu(\boldsymbol{\theta}_{(i)})$ and $\mathrm{sd}(\boldsymbol{\theta}_{(i)})$, $i = 1, \ldots, N_s$, where $N_s = 20$. Then we find that the following values

$$E\left[\mu(\boldsymbol{\theta}_{(i)}^{\mathrm{SKF}}) - \mu(\boldsymbol{\theta}_{(i)}^{\mathrm{AugMCMC}})\right], \qquad E\left[\mathrm{sd}(\boldsymbol{\theta}_{(i)}^{\mathrm{SKF}}) - \mathrm{sd}(\boldsymbol{\theta}_{(i)}^{\mathrm{AugMCMC}})\right], \tag{4.18}$$

where the expectation is taken on i, are non-distinguishable among the three SKF schemes, which means the posterior densities produced by the three SKF schemes are very close.

Figure 4.9 shows the posterior marginal densities of two randomly selected datasets from the 20 2-asset simulated datesets. In the top plot of Figure 4.9, we can see the posterior

densities of ρ_1 and τ_1 from the 30-particle scheme are a little deviated from the other densities from the 100-particle and 300-particle schemes. However, the posterior densities of the three schemes are generally very close. We also check the quantities of (4.18) for the $\boldsymbol{\theta}$ of this 2-asset model, and find them to be very similar among the three scheme. It means that the 30-particle scheme obtain similar posterior densities to the 300-particle scheme not only in a single-asset model analysis but also in a 2-asset model analysis.

Although using more particles in the SKF method can lead to better filtering mean and variance estimations, it has no obvious advantage on the posterior parameter estimation based on the posterior results from 2 models each with $N_s = 20$ datasets. The posterior results suggest using $M = 30$ particles can obtain comparable results to using $M = 300$ particles at least for the single-asset and the 2-asset eOU models.

In summary of this Section 4.5, the approximations we made in the SKF and MQ methods have an impact on the approximated posterior densities such that they deviate slightly from their exact densities. Due to the differences are very small, we find that the numerical evidence generally supports the Gaussian assumptions of the SKF method by studying the two eOU models. We also find that the number of particles of $M = 30$ is comparable to the number of particles of $M = 300$ in estimating the posterior density, and the estimated posterior densities very close to those of the AugMCMC benchmark. Upon those findings, we believe that the SKF method implemented by the MQ algorithm 4.4 is one competitive method of estimating the parameters in the eOU models due to the fast computational speed and the reasonable estimation accuracy.

4.6 Comparisons with Existing Methods

In this section, we study several parameter estimation methods in a large-scale simulation experiment on the multi-asset eOU model (4.16). We begin this section by explaining some candidate methods, then show the results of the simulation study result. Since we have extensively explained the AugMCMC method, which is detailed in Appendix B, and our proposed SKF method combining with the MQ algorithm 4.4, we will not further explain them in the following.

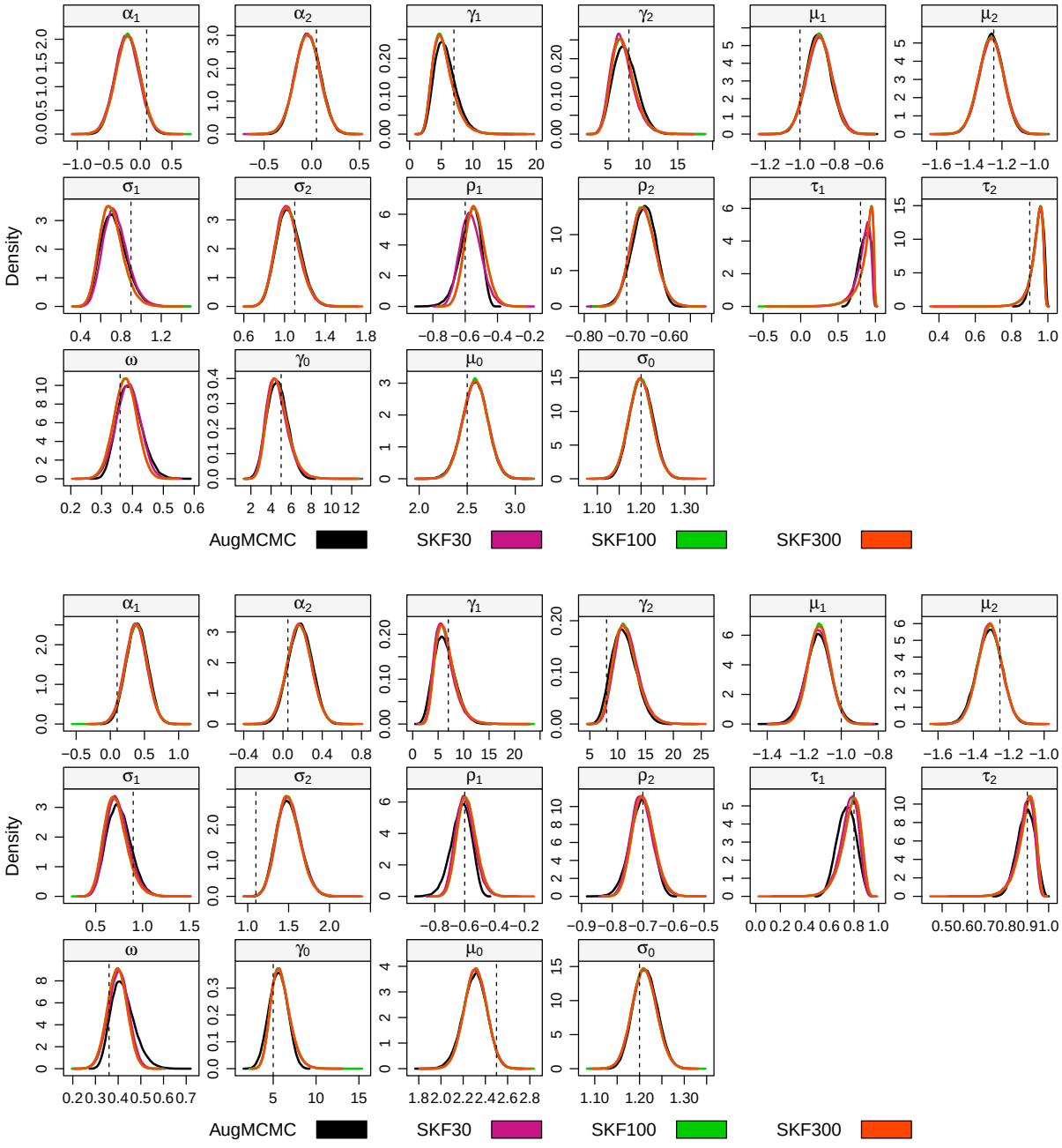


Figure 4.9: Posterior density comparison using different number of particles on two randomly selected datasets of 2-asset model. *Top:* the first dataset; *bottom:* the second dataset.

4.6.1 Unscented Kalman Filter for eOU model

Originally proposed by Julier and Uhlmann (1997), the Unscented Kalman Filter (UKF) exploits the fact that it should not require a large number of MC simulations to estimate the first two moments of an nonlinearly transformed random variable/vector, given that we know its original first two moments. In fact, the UKF uses a set of carefully chosen so-called sigma points around the original mean, then propagates those points by the nonlinear function. Applying a set of deterministic weights, we can obtain the weighted mean and variance of those propagated points. This set of estimated mean and variance is proven to be accurate up to the second-order of the Taylor expansion for the conventional state-space model (Julier and Uhlmann, 1997). Note that the number of the sigma points of the UKF method is $2q + 1$, where q is the dimension of the latent variables. It is a remarkably small set of sample, which shall cut down the computational time significantly. The UKF method has typically been used for simplified Gaussian state-space models with constant $\boldsymbol{\Omega_\theta}$. There exist a large body of studies conclude that it is a highly accurate and powerful tool for nonlinear filtering problems (e.g., LaViola, 2003; Van Der Merwe et al., 2004a,b; Xiong et al., 2006; Plett, 2006; Sarkka, 2007; Kandepu et al., 2008).

We adopt a slightly different implementation (Wan and Van Der Merwe, 2000) and describe the details of the algorithm in the framework of the GG-SSM (4.2) in Appendix C. To the best of our knowledge, this implementation for arbitrary GG-SSMs (4.2) has not been discussed before. We use the UKF to filter through the latent volatility of one simulated dataset from the single-asset eOU model (4.15) using the set of parameters in Table 4.2. The same dataset is filtered by the SKF method and the particle filter as well for comparison. In Figure 4.10, we show the 95% credible intervals of the filtering densities from the three filtering methods. The particle filter and the SKF method have very similar filtering densities. However, the UKF has significantly narrower credible intervals and the filtering means are not very close to the ones from the other two methods. Further, we can find the posterior density by the UKF method combining with the MQ algorithm 4.4, but the posterior results are quite far from the benchmark results. Hence, we skip the result here.

By our experiment, the UKF's accuracy cannot be guaranteed when $\boldsymbol{\Omega_\theta}$ is not constant

but depends on the model variables. In conclusion, we find that the UKF is not suitable for this study and we shall exclude the UKF in the later simulation study.

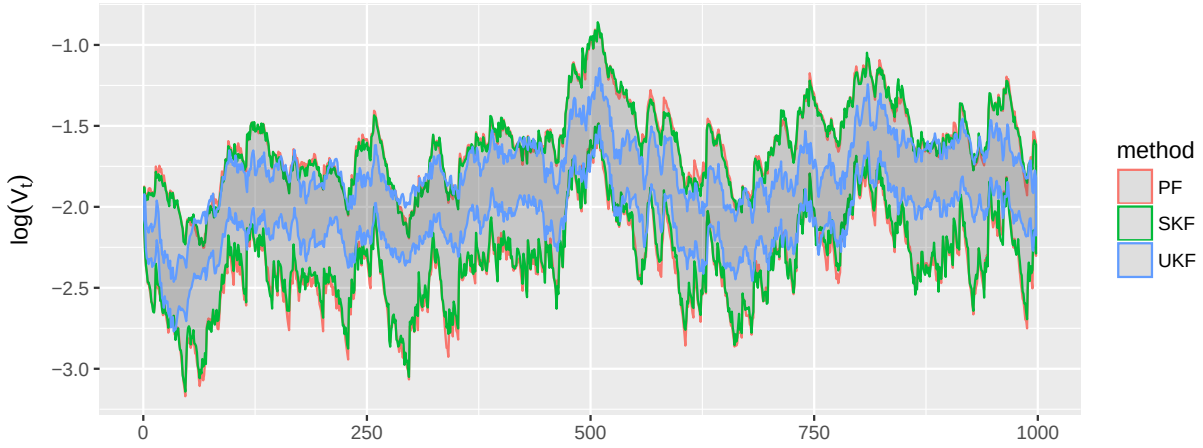


Figure 4.10: Comparison of 95% credible intervals of filtering densities of the volatilities.

4.6.2 The Laplace Approximation Method

The Laplace approximation (Laplace, 1774) is a numerical method to approximate the integral of the form $\int_a^b \exp(g(x))\,\mathrm{d}x$, where $g(x)$ is some twice-differentiable function. This method can be conveniently applied to approximate the likelihood function (4.3). That is, instead of filtering through the latent variables step by step, this method applies the Laplace approximation on the integral of

$$p(\boldsymbol{X}_{1:n}|\boldsymbol{\theta}) = \int p(\boldsymbol{X}_{1:n}, \boldsymbol{Z}_{1:n}|\boldsymbol{\theta}) d\boldsymbol{Z}_{1:n}.$$

Firstly, we approximate $p(\boldsymbol{X}_{1:n}, \boldsymbol{Z}_{1:n}|\boldsymbol{\theta})$ as a multivariate Gaussian density function on $\boldsymbol{Z}_{1:n}$ marginally. The mean $\hat{\boldsymbol{Z}}_{1:n}$ and variance-covariance matrix $\boldsymbol{\Sigma}$ of this multivariate Gaussian density are obtained by $\hat{\boldsymbol{Z}}_{1:n} = \arg\max_{\boldsymbol{Z}_{1:n}} p(\boldsymbol{X}_{1:n}, \boldsymbol{Z}_{1:n}|\boldsymbol{\theta})$ and $\boldsymbol{\Sigma}$ is the inverse negative Hessian matrix of $\boldsymbol{Z}_{1:n}$ with respect to $p(\boldsymbol{X}_{1:n}, \boldsymbol{Z}_{1:n}|\boldsymbol{\theta})$ evaluated on the point of $\hat{\boldsymbol{Z}}_{1:n}$. Hence, the model likelihood approximation is

$$p_{Laplace}(\boldsymbol{X}_{1:n}|\boldsymbol{\theta}) = |\det(\boldsymbol{\Sigma})|^{-1/2} p(\boldsymbol{X}_{1:n}, \hat{\boldsymbol{Z}}_{1:n}|\boldsymbol{\theta}). \tag{4.19}$$

Apparently, if $p(\boldsymbol{X}_{1:n}, \boldsymbol{Z}_{1:n}|\boldsymbol{\theta})$ is Gaussian density, e.g., the linear model (4.8), then this method is in fact exact. If not, the closer the above densities are to their Gaussian approximations, the better is this Laplace approximation.

For the (multi-asset) eOU model, the smoothing densities $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_t|\boldsymbol{X}_{1:n})$, $t = 1, \ldots, n$ of the eOU model should be very close to Gaussian distribution. Hence, we believe that $p(\boldsymbol{Z}_{1:n}|\boldsymbol{X}_{1:n}, \boldsymbol{\theta})$ should also be very close to a multivariate Gaussian distribution, which fits to the assumption of the Laplace approximation. In addition, the surface of $p(\boldsymbol{X}_{1:n}, \boldsymbol{Z}_{1:n}|\boldsymbol{\theta})$ should be rather convex on $\boldsymbol{Z}_{1:n}$ marginally, so that finding $\hat{\boldsymbol{Z}}_t$ can be reasonably handled by the Newton's method. In summary, the evaluation of the approximate likelihood (4.19) is relatively cheap.

Recently, there is an interesting development on the numerical implementation of the Laplace method for GG-SSM (4.2). The R package *TMB* (Kristensen et al., 2016) combines the Laplace approximation method with the automatic differentiation implemented by the package *CppAD* (Bell, 2012), which derives the gradient of a function by chain rule over its elementary arithmetic operations (e.g., addition and multiplication) and elementary functions (e.g., exponential and logarithm). It uses a Newton's method to find $\hat{\boldsymbol{Z}}_{1:n}$ and the AD offers fast evaluations of the first and second derivatives of the approximated likelihood (4.19) to the model parameters. The user needs to contribute a C++ code for the likelihood/posterior evaluation. A Bayesian inference can be conducted by the MCMC on the approximate posterior

$$p_{Laplace}(\boldsymbol{\theta}|\boldsymbol{X}_{1:n}) \propto \pi(\boldsymbol{\theta}) p_{Laplace}(\boldsymbol{X}_{1:n}|\boldsymbol{\theta})$$

using an associated R package *tmbstan* (Kristensen, 2017). We attempt this MCMC algorithm and find that it is relatively slow due to that a number of likelihood evaluations is required to obtain one sample by the Hamiltonian Monte Carlo (Duane et al., 1987).

Note that this Laplace method offers the approximate likelihood as a direct function of the model parameters. In addition, the gradient function from the AD offers an extremely fast way to solve the problem of maximum likelihood/posterior density. Hence, the MQ algorithm 4.4 can also be combined with this Laplace method to perform an approximate posterior inference. Indeed, we apply this Laplace method with the MQ algorithm 4.4 in the later simulation study.

4.6.3 Simulation Study

The simulation study is based on a 5-asset eOU model (4.16). Based on the true parameters as shown in Table 4.5, we simulate 200 paths, 1000 daily observations each ($\Delta T = \frac{1}{252}$). When analyzing these datasets, we treat the volatility factor and the asset prices as observed and the other volatilities as latent. In other words, we are dealing with a 5-dimensional latent variables problem.

α_1	α_2	α_3	α_4	α_5	γ_1	γ_2	γ_3	γ_4	γ_5	μ_1	μ_2	μ_3	μ_4
0.1	0.05	0.0	-0.05	-1	7.0	8.0	6.0	7.0	8.0	-1.0	-1.25	-1.1	-0.9
μ_5	σ_1	σ_2	σ_3	σ_4	σ_5	ρ_1	ρ_2	ρ_3	ρ_4	ρ_5	τ_1	τ_2	τ_3
-1.0	0.9	1.1	1.0	1.2	1.0	-0.6	-0.7	-0.65	-0.75	-0.7	0.8	0.9	0.8
τ_4	τ_5	ω_1	ω_2	ω_3	ω_4	ω_5	γ_0	μ_0	σ_0				
0.85	0.8	0.6	0.7	0.6	0.65	0.7	5	2.5	1.2				

Table 4.5: True parameters of the 5-asset eOU model.

In Subsection 4.6.1, we rule out the UKF method. Hence, we analyze the 200 simulated datasets from the 5-asset eOU model using the three methods, i.e., the AugMCMC method, the SKF method and the Laplace method. We focus on comparing the posterior parameter sample from those three methods. A posterior sample size of 1 million will be used for all of the methods. For the SKF method, we use 30 particles. Both of the SKF method and the Laplace method draw approximate posterior samples via the MQ Algorithm 4.4.

In Figure 4.11, we show the marginal posterior densities of all of the parameters on one randomly selected dataset. For some parameters, i.e., σ_4, ρ_4, ρ_5, τ_5 and ω_4, the Laplace method is closer to the AugMCMC benchmark, in terms of the modes of the posterior densities. For some parameters, i.e., γ_0, ω_5, both of the two approximate methods are a little off the benchmark densities. Generally speaking, the posterior densities are very close among the three methods.

In Table 4.6, we find that the average relative errors for some selected parameters in

the 5-asset eOU model (4.16). The relative error is calculated as

$$E\left[\frac{\mu(\boldsymbol{\theta}_{(i)}^{\text{SKF}}) - \mu(\boldsymbol{\theta}_{(i)}^{\text{AugMCMC}})}{\mu(\boldsymbol{\theta}_{(i)}^{\text{AugMCMC}})}\right], \qquad E\left[\frac{\mu(\boldsymbol{\theta}_{(i)}^{\text{Laplace}}) - \mu(\boldsymbol{\theta}_{(i)}^{\text{AugMCMC}})}{\mu(\boldsymbol{\theta}_{(i)}^{\text{AugMCMC}})}\right],$$

which adopts the notations of (4.18) for $i = 1, \dots 200$. As mentioned in Section 4.4.1 that we operate on the unbounded version of the parameters in practice, so the results in Table 4.4 are calculated using the unbounded version, except of σ_i, $i = 1, \dots, q$. Their average relative errors are calculated on the regular scale since their unbounded versions are very close to 0, which makes the relative errors reflect unreasonable results. Due to the same reasoning of being close to 0, α_i, $i = 1, \dots, q$, are excluded from this table. The results in Table 4.6 show that both approximation methods estimate the modes of the posterior reasonably close to the benchmarks. Moreover, the Laplace approximation consistently estimate a little better than the SKF method except for γ_0. This suggests that $p(\boldsymbol{Z}_{1:n}|\boldsymbol{X}_{1:n}, \boldsymbol{\theta})$ of the multi-asset eOU model (4.16) should be very close to a multivariate Gaussian distribution, so that the results from the Laplace method and the SKF methods are very close to the benchmark. However, the SKF is limited by the extra assumption of the $p_{\boldsymbol{\theta}}(\boldsymbol{X}_t|\boldsymbol{X}_{1:t-1})$, $t = 1, \dots, n$, being Gaussian, which is not required by the Laplace method.

Percentage	γ_1	γ_2	γ_3	γ_4	γ_5	μ_1	μ_2	μ_3	μ_4	μ_5	σ_1
SKF	2.64	1.79	2.77	2.03	2.42	1.09	0.81	1.13	1.47	1.09	3.70
Laplace	1.87	1.45	2.21	1.86	1.38	0.64	0.57	0.59	0.84	0.63	1.26
	σ_2	σ_3	σ_4	σ_5	ρ_1	ρ_2	ρ_3	ρ_4	ρ_5	τ_1	τ_2
SKF	2.23	3.26	2.66	3.56	2.77	1.61	2.35	1.52	2.1	3.38	2.95
Laplace	0.94	0.87	0.62	1.26	1.91	1.01	1.61	1.12	1.58	1.45	1.44
	τ_3	τ_4	τ_5	ω_1	ω_2	ω_3	ω_4	ω_5	γ_0	μ_0	σ_0
SKF	2.78	1.83	2.27	2.24	2.27	2.46	2.53	3.75	1.88	0.59	1.64
Laplace	0.9	0.59	0.92	1.92	1.87	2.04	2.15	3.39	2.61	0.51	1.62

Table 4.6: Average relative error of selected parameters.

Table 4.7 compares the coverages of 50%, 75% and 95% credible intervals of the posterior parameter densities to their nominal values. That is, the table reports the fraction of 200

datasets for which the true parameter was inside the posterior credible intervals of the probabilities of 50%, 75% and 95%. In general, all of the results from the three competing models are very close to each other. According to the 95% credible intervals results, the approximate methods often undercover, which suggests their standard deviations are too small. Table 4.7 also reports the average root mean square errors (RMSE) over the 200 RMSEs, each of which is calculated as $\sqrt{E\left(\theta_{\text{post}} - \theta_{\text{true}}\right)^2}$, where θ represents an arbitrary parameter in the 5-asset eOU model and the expectation is taken on the posterior sample. The AugMCMC method does not have an significant advantage over the others based on the RMSE results.

In summary, the posterior densities of the approximate methods in this simulation study for the 5-asset eOU model (4.16) are reasonably close to those of the AugMCMC method. Often, the posterior samples from the approximate methods produce slightly narrower credible intervals than those of the AugMCMC method. In terms of the average RMSE of the posterior samples, all three models perform similarly.

	50% Coverage			75% Coverage			95% Coverage			RMSE		
	A [a]	S [a]	L [a]	A	S	L	A	S	L	A	S	L
α_1	0.54	0.56	0.55	0.75	0.76	0.76	0.96	0.95	0.96	0.23	0.24	0.24
α_2	0.52	0.47	0.50	0.79	0.76	0.78	0.96	0.95	0.95	0.18	0.18	0.18
α_3	0.45	0.43	0.46	0.69	0.66	0.68	0.94	0.92	0.92	0.22	0.23	0.22
α_4	0.49	0.46	0.46	0.70	0.69	0.71	0.94	0.93	0.93	0.26	0.26	0.26
α_5	0.45	0.46	0.43	0.69	0.68	0.68	0.95	0.94	0.94	0.24	0.25	0.25
γ_1	0.35	0.37	0.34	0.57	0.54	0.52	0.83	0.78	0.78	4.90	4.53	4.84
γ_2	0.35	0.35	0.29	0.55	0.54	0.49	0.78	0.78	0.75	3.78	3.59	3.89
γ_3	0.35	0.32	0.28	0.54	0.51	0.47	0.83	0.77	0.74	3.53	3.31	5.92
γ_4	0.34	0.33	0.30	0.57	0.51	0.52	0.85	0.77	0.78	3.02	2.87	3.19
γ_5	0.30	0.34	0.26	0.58	0.57	0.53	0.82	0.78	0.77	4.42	4.11	4.49
μ_1	0.50	0.47	0.48	0.73	0.70	0.69	0.92	0.92	0.91	0.13	0.08	0.08
μ_2	0.50	0.46	0.47	0.69	0.69	0.68	0.94	0.92	0.92	0.13	0.09	0.09
μ_3	0.52	0.47	0.50	0.67	0.66	0.65	0.92	0.86	0.88	0.13	0.10	0.10
μ_4	0.46	0.41	0.42	0.74	0.72	0.70	0.93	0.91	0.92	0.14	0.10	0.10

μ_5	0.46	0.44	0.45	0.68	0.66	0.67	0.93	0.90	0.92	0.12	0.08	0.08
σ_1	0.55	0.52	0.54	0.78	0.72	0.77	0.92	0.90	0.91	0.22	0.22	0.22
σ_2	0.46	0.42	0.46	0.69	0.66	0.68	0.94	0.91	0.93	0.19	0.19	0.19
σ_3	0.44	0.43	0.44	0.70	0.65	0.70	0.92	0.90	0.90	0.21	0.20	0.25
σ_4	0.45	0.42	0.46	0.72	0.65	0.71	0.94	0.90	0.95	0.19	0.19	0.19
σ_5	0.49	0.41	0.44	0.72	0.66	0.71	0.92	0.90	0.92	0.21	0.21	0.22
ρ_1	0.56	0.52	0.57	0.76	0.76	0.78	0.96	0.95	0.96	0.07	0.07	0.07
ρ_2	0.45	0.44	0.45	0.69	0.70	0.72	0.95	0.92	0.94	0.05	0.05	0.05
ρ_3	0.47	0.46	0.50	0.70	0.69	0.68	0.94	0.90	0.94	0.07	0.06	0.07
ρ_4	0.48	0.47	0.49	0.76	0.71	0.76	0.94	0.92	0.96	0.04	0.04	0.05
ρ_5	0.51	0.49	0.50	0.73	0.72	0.76	0.95	0.94	0.94	0.06	0.06	0.06
τ_1	0.52	0.50	0.51	0.73	0.70	0.73	0.97	0.93	0.93	0.08	0.08	0.08
τ_2	0.44	0.40	0.43	0.68	0.69	0.71	0.94	0.91	0.92	0.05	0.05	0.05
τ_3	0.46	0.40	0.45	0.70	0.64	0.68	0.92	0.88	0.90	0.07	0.07	0.08
τ_4	0.54	0.52	0.58	0.73	0.71	0.74	0.94	0.90	0.93	0.05	0.05	0.05
τ_5	0.50	0.47	0.50	0.78	0.73	0.78	0.95	0.93	0.95	0.06	0.06	0.06
ω_1	0.48	0.46	0.49	0.73	0.74	0.74	0.94	0.98	0.98	0.06	0.06	0.06
ω_2	0.49	0.48	0.50	0.76	0.77	0.79	0.94	0.96	0.96	0.06	0.05	0.05
ω_3	0.43	0.47	0.48	0.72	0.73	0.75	0.92	0.94	0.95	0.07	0.06	0.06
ω_4	0.50	0.48	0.51	0.76	0.76	0.79	0.95	0.95	0.96	0.07	0.06	0.06
ω_5	0.56	0.55	0.57	0.79	0.80	0.80	0.96	0.94	0.95	0.07	0.06	0.06
γ_0	0.28	0.26	0.20	0.48	0.45	0.35	0.83	0.72	0.67	1.61	1.64	1.78
μ_0	0.43	0.42	0.40	0.68	0.66	0.64	0.92	0.90	0.89	0.20	0.14	0.14
σ_0	0.52	0.52	0.52	0.76	0.74	0.74	0.94	0.90	0.90	0.04	0.04	0.04

Table 4.7: Coverage and RMSE of parameters.
A = AugMCMC, S = SKF method, L = Laplace method

4.6.4 Computational Complexity

In the previous subsection, we find that the exact posterior densities of the AugMCMC method and the approximate posterior densities of the SKF method and the Laplace

method are close. Hence, the computational cost becomes very important. In the following, we compare the computational order of those three methods.

1. The AugMCMC method: according to its detailed algorithm in Appendix B, we can summarize its computational complexity. For each update of latent volatilities $V_t = (V_{1,t}, \cdots, V_{q,t})$, the main computation involves one simulation from a proposal q-dimensional Gaussian distribution and two evaluations of $(p+q)$-dimensional Gaussian distribution. Hence, the computational order of updating all the latent variables is $\mathcal{O}(n(p+q)^3)$. Via the vanilla Metropolis-Hastings algorithm, updating one parameter needs to evaluate the full posterior density $p(\boldsymbol{Y}_{1:t}|\boldsymbol{\theta})\pi(\theta)$, where the full data likelihood $p(\boldsymbol{Y}_{1:t}|\boldsymbol{\theta})$ needs n sequential evaluation of $p_{\boldsymbol{\theta}}(\boldsymbol{Y}_t|\boldsymbol{Y}_{1:t-1})$. Denote the number of the parameters as r, then a computational order of $\mathcal{O}(rn(p+q)^3)$ is required to update all of the parameters. In total, each update of the latent variables and the parameters requires the computational complexity of $\mathcal{O}((r+1)n(p+q)^3)$.

2. The SKF method: according to Algorithm 4.3, for each update of the filtering distribution $p_{\boldsymbol{\theta}}(\boldsymbol{Z}_{t+1}|\boldsymbol{X}_{1:t})$, we use M particles. For each particle, it mainly involves one q-dimensional Gaussian simulation, one p-dimensional Gaussian density evaluation and one p-dimensional matrix inversion. Hence, the computational complexity is $\mathcal{O}(M(q^3+p^3))$ for each updating. Since we need to update n times, which is the length of the observations, the total computational complexity of one evaluation of the likelihood is $\mathcal{O}(MN(q^3+p^3))$.

3. The Laplace method: the R package *TMB* has an internal optimization algorithm to find $\hat{\boldsymbol{Z}}_{1:n} = \arg\max_{\boldsymbol{Z}_{1:n}} p(\boldsymbol{X}_{1:n}, \boldsymbol{Z}_{1:n}|\boldsymbol{\theta})$. We contribute the C++ code to find the full data likelihood $p(\boldsymbol{X}_{1:n}, \boldsymbol{Z}_{1:n}|\boldsymbol{\theta})$, which is obtained by the product of transition densities $p_{\boldsymbol{\theta}}(\boldsymbol{Y}_t|\boldsymbol{Y}_{t-1})$, so that it has a computational order of $\mathcal{O}(N(p+q)^3)$. While it is hard to determine computational order of the iterations needed to find $\hat{\boldsymbol{Z}}_{1:n}$.

Upon the results above, the exact computational speeds of the three methods are still obscured. In order to clarify, we have to answer the questions like: how many MCMC iterations are necessary for the AugMCMC method; how may likelihood evaluations are

needed by the optimizer to find the MAP for the SKF method; What is computational order to find $\hat{\boldsymbol{Z}}_{1:n}$ by the Laplace method. Those are the questions that depend on the practical situations and often lack answers. Instead, we compare the actual CPU time. All of the main computations of the three methods are coded in C++. Although the code qualities might vary, we do not expect a huge difference. Under the same computational environment, we record the rough CPU time in Table 4.8. The CPU time for the AugMCMC method is the CPU time to draw 1 million MCMC samples; the CPU time for the SKF method and the Laplace method is the CPU time to solve the MAP and the numeric Hessian matrix as required in the MQ algorithm 4.4. 30 particles is used for the SKF method.

	AugMCMC	SKF + MQ	Laplace + MQ
1-asset model	30 min	20 sec	15 sec
2-asset model	5 hr	2 min	3 min
5-asset model	50 hr	40 min	1 hr

Linux server with CPU: Intel Xeon E5-4660v3 2.1 GHz

Table 4.8: The comparison of CPU time.

The results in Table 4.8 show that the MQ algorithm 4.4 cuts down the CPU time dramatically when compared to the AugMCMC method. In particle, the SKF CPU time is consistently around only 1% of the CPU time of the AugMCMC method. Between the SKF method and the Laplace method, the CPU time of the SKF grows slower than the Laplace method as the model dimension grows. One thing is clear from the discussion of the computational orders that the computational burden of the SKF method grows much slower than the other two methods as the dimension of the model (p and q) grows. The computational complexities of the AugMCMC method and the Laplace method are in the ratio of $(p+q)^3$ but the computational complexity of the SKF method is in the ratio of (p^3+q^3), which makes a big difference when p and q are large.

4.7 Conclusion

In this chapter, we introduce the Synthetic Kalman Filter and examine it using the multi-asset eOU SV model. The Gaussian assumptions in the SKF are assessed to be reasonable under the eOU models. Using QMC, 30 particle is sufficient for the SKF, which cuts down the computational cost dramatically. The simulation study of 200 datasets from the 5-asset eOU model shows that the computational gain of the approximation methods, i.e., the SKF method and the Laplace method, is huge but the loss of the accuracy of the posterior density is not significant. In particular, the computational order of the SKF method grows slower than the competing methods as the dimension of the model variables grows.

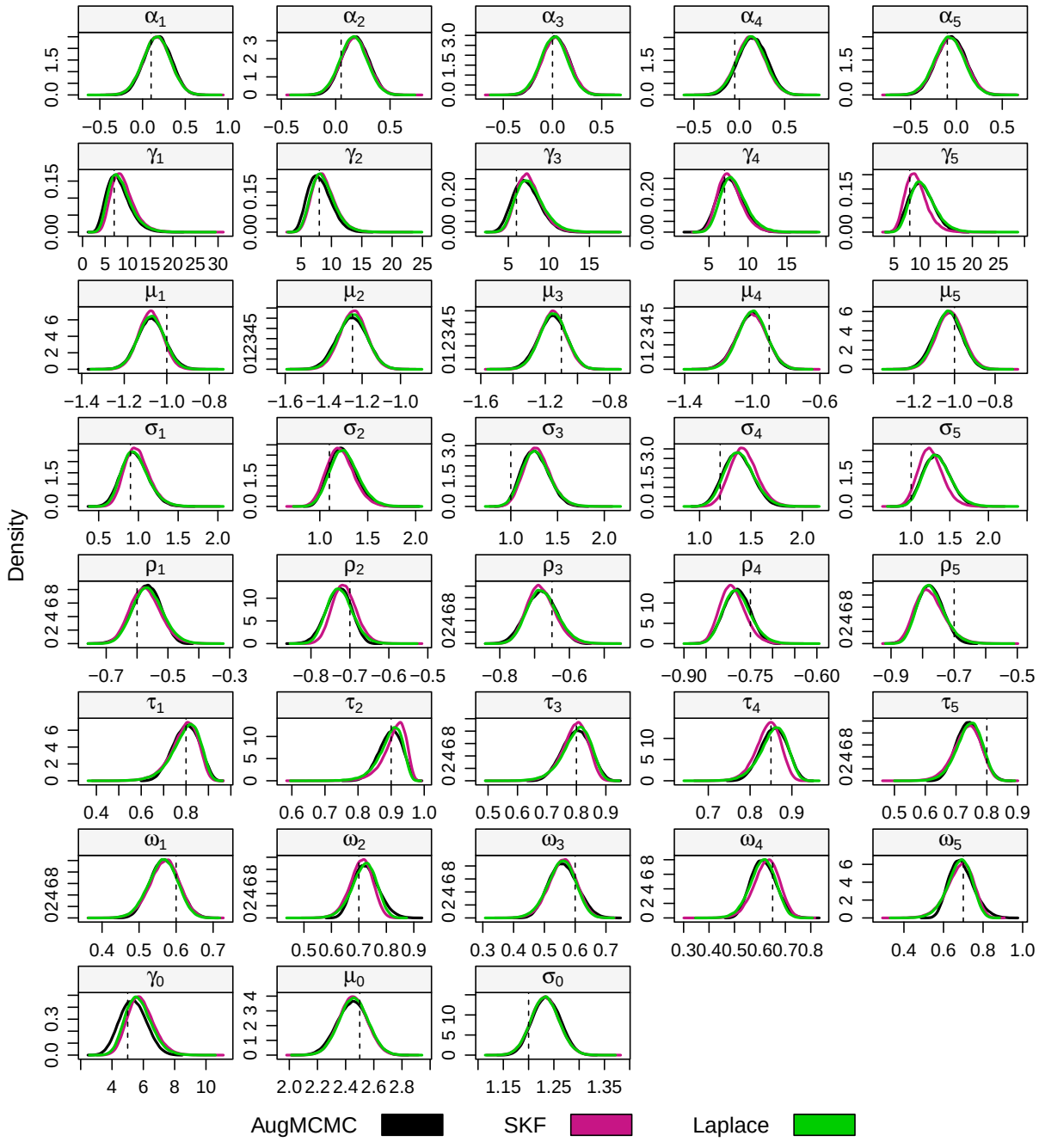


Figure 4.11: Comparison of the posterior densities on one randomly selected 5-asset dataset.

Chapter 5

Conclusion

In this thesis, we focus on the stochastic volatility modeling, especially the multi-asset modeling, and its inference methods.

First of all, we target the high computational cost of the parameter estimation of the stochastic volatility models. In Chapter 2, we propose to use an observed volatility proxy instead of treating the volatility as latent so that the model likelihood is based on a fully observed dataset. Hence, the inference becomes simple and fast. In Chapter 3, we propose a parsimonious multi-asset SV model so that we avoid too many parameters when the number of the assets grows. In Chapter 4, we propose a fast filtering method which is able to approximate the model likelihood extremely close to its true but significantly reduce the computational cost by around two orders of magnitude.

Second, we compare several models, including the Heston model and the exponential OU model, by their out-of-sample forecasts of the asset prices and option prices. In Chapter 2, we propose a model that is comparable to the Heston model on both long-term and short-term forecasts of the SPX. In Chapter 3, the multi-asset model we propose is better than the benchmark multi-asset exponential OU model for option forecasting and almost equivalent for asset price forecasting.

Third, we explore the potential usage of the VIX in stochastic volatility modeling. In Chapter 2, we replace the latent volatility of the SPX directly by the VIX and find its

forecasts similar to the Heston model. While the VIX is not a good candidate to replace the volatilities of some individual stocks, since the dynamics of their volatility can be different to the market volatility. In Chapter 3, we use the VIX as the common volatility factor of the latent volatilities of the individual stocks and find that the model performs very well for option pricing. As a result, it makes more sense to use the VIX as auxiliary information such as a volatility factor than to use it as a direct replacement for the latent volatility.

Finally, we compare several parameter estimation methods in Chapter 4 under a generalized Gaussian state-space model framework. An exact MCMC method might be unnecessary given that the approximation methods are extremely close to it for a 5-asset eOU SV model. Comparing the methods, i.e., the proposed SKF method and the Laplace method, the computational cost of the SKF method scales better as the number of the model variables grows.

In summary, new modeling frameworks and new inference methods are proposed for the multi-asset stochastic volatility modeling in order to find computationally more efficient approaches. By comparing to the state-of-art models and inference methods, we find the proposed works provides improvements and can potentially be generalized into other applications. Given that the results are similar or even better, our approaches reduce the computational time significantly.

References

Ahoniemi, K. (2006). Modeling and Forecasting Implied Volatility: An Econometric Analysis of the VIX Index. *Helsinki: Helsinki Center of Economic Research*.

Ait-Sahalia, Y., Fan, J., and Li, Y. (2013). The Leverage Effect Puzzle: Disentangling Sources of Bias at High Frequency. *Journal of Financial Economics*, 109(1):224–249.

Aït-Sahalia, Y. and Kimmel, R. (2007). Maximum Likelihood Estimation of Stochastic Volatility Models. *Journal of Financial Economics*, 83(2):413–452.

Alizadeh, S., Brandt, M. W., and Diebold, F. X. (2002). Range-based Estimation of Stochastic Volatility Models. *The Journal of Finance*, 57(3):1047–1091.

Andersen, L. B. (2007). Efficient Simulation of the Heston Stochastic Volatility Model. Technical report, Bank of America Securities.

Andersen, T. G. and Bollerslev, T. (1998). Answering the Skeptics: Yes, Standard Volatility Models Do Provide Accurate Forecasts. *International economic review*, 39:885–905.

Andersen, T. G., Bollerslev, T., Diebold, F. X., and Labys, P. (2003). Modeling and forecasting Realized Volatility. *Econometrica*, 71(2):579–625.

Andersen, T. G. and Lund, J. (1997). Estimating Continuous-time Stochastic Volatility Models of the Short-term Interest Rate. *Journal of econometrics*, 77(2):343–377.

Anderson, B. D. and Moore, J. B. (1979). Optimal Filtering. *Englewood Cliffs*, 21:22–95.

Ando, T., Zellner, A., et al. (2010). Hierarchical Bayesian Analysis of the Seemingly Unrelated Regression and Simultaneous Equations Models Using A Combination of Direct Monte Carlo and Importance Sampling Techniques. *Bayesian Analysis*, 5(1):65–95.

Andrieu, C. and Doucet, A. (2002). Particle filtering for partially Observed Gaussian State Space Models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(4):827–836.

Andrieu, C., Doucet, A., and Holenstein, R. (2010). Particle Markov Chain Monte Carlo Methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(3):269–342.

Andrieu, C., Roberts, G. O., et al. (2009). The Pseudo-marginal Approach for Efficient Monte Carlo Computations. *The Annals of Statistics*, 37(2):697–725.

Arasaratnam, I. and Haykin, S. (2009). Cubature Kalman Filters. *IEEE Transactions on automatic control*, 54(6):1254–1269.

Asai, M. and McAleer, M. (2009). The Structure of Dynamic Correlations in Multivariate Stochastic Volatility Models. *Journal of Econometrics*, 150(2):182–192.

Asai, M., McAleer, M., and Yu, J. (2006). Multivariate Stochastic Volatility: A Review. *Econometric Reviews*, 25(2-3):145–175.

Aydemir, A. B. (2002). Volatility Modelling in Finance. *Forecasting Volatility in the Financial Markets*.

Barndorff-Nielsen, O. E. (1997). Normal Inverse Gaussian Distributions and Stochastic Volatility Modelling. *Scandinavian Journal of statistics*, 24(1):1–13.

Bates, D. S. (2006). Maximum Likelihood Estimation of Latent Affine Processes. *The Review of Financial Studies*, 19(3):909–965.

Bell, B. M. (2012). Cppad: a Package for c++ Algorithmic Differentiation. *Computational Infrastructure for Operations Research*, 57.

Bengtsson, T., Bickel, P., Li, B., et al. (2008). Curse-of-dimensionality Revisited: Collapse of the Particle Filter in Very Large Scale Systems. *Probability and statistics: Essays in honor of David A. Freedman*, 2:316–334.

Beskos, A., Kalogeropoulos, K., and Pazos, E. (2013). Advanced MCMC methods for Sampling on Diffusion Pathspace. *Stochastic Processes and their Applications*, 123(4):1415–1453.

Black, F. (1975). Fact and Fantasy in the Use of Options. *Financial Analysts Journal*, 31:36–72.

Black, F. and Scholes, M. (1973). The Pricing of Options and Corporate Liabilities. *The journal of political economy*, 3:637–654.

Bladt, M., Finch, S., and Sørensen, M. (2016). Simulation of Multivariate Diffusion Bridges. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(2):343–369.

Blair, B. J., Poon, S.-H., and Taylor, S. J. (2001). Forecasting S&P 100 Volatility: the Incremental Information Content of Implied Volatilities and High-frequency Index Returns. *Journal of Econometrics*, 105:5–26.

Bollerslev, T. (1986). Generalized Autoregressive Conditional Heteroskedasticity. *Journal of econometrics*, 31(3):307–327.

Bollerslev, T. (1987). A Conditionally Heteroskedastic Time Series Model for Speculative Prices and Rates of Return. *The review of economics and statistics*, 69:542–547.

Bollerslev, T. (1990). Modelling the Coherence in Short-run Nominal Exchange Rates: A Multivariate Generalized ARCH Model. *The review of economics and statistics*, 72:498–505.

Bollerslev, T., Chou, R. Y., and Kroner, K. F. (1992). Arch Modeling in Finance: A Review of the Theory and Empirical Evidence. *Journal of econometrics*, 52(1-2):5–59.

Bouchaud, J.-P., Matacz, A., and Potters, M. (2001). Leverage Effect in Financial Markets: The Retarded Volatility Model. *Physical review letters*, 87(22):228701.

Broto, C. and Ruiz, E. (2004). Estimation Methods for Stochastic Volatility Models: A Survey. *Journal of Economic Surveys*, 18(5):613–649.

Carr, P., Geman, H., Madan, D. B., and Yor, M. (2003). Stochastic Volatility for Lévy Processes. *Mathematical Finance*, 13(3):345–382.

Chan, D., Kohn, R., and Kirby, C. (2006). Multivariate Stochastic Volatility Models with Correlated Errors. *Econometric Reviews*, 25(2-3):245–274.

Cisana, E., Fermi, L., Montagna, G., and Nicrosini, O. (2007). A Comparative Study of Stochastic Volatility Models. Technical report.

Corradi, V. (2000). Reconsidering the Continuous Time Limit of the GARCH (1, 1) Process. *Journal of Econometrics*, 96(1):145–153.

Cox, J. C., Ingersoll Jr, J. E., and Ross, S. A. (1985). A Theory of the Term Structure of Interest Rates. *Econometrica: Journal of the Econometric Society*, 53:385–407.

Cox, J. C., Ross, S. A., Rubinstein, M., et al. (1979). Option Pricing: A Simplified Approach. *Journal of financial Economics*, 7(3):229–263.

Doucet, A., Godsill, S., and Andrieu, C. (2000). On Sequential Monte Carlo Sampling Methods for Bayesian Filtering. *Statistics and computing*, 10(3):197–208.

Doucet, A. and Johansen, A. M. (2009). A Tutorial on Particle Filtering and Smoothing: Fifteen Years Later. *Handbook of nonlinear filtering*, 12(656-704):3.

Drost, F. C. and Werker, B. J. (1996). Closing the GARCH Gap: Continuous Time GARCH Modeling. *Journal of Econometrics*, 74(1):31–57.

Duan, J.-C. (1995). The GARCH Option Pricing Model. *Mathematical finance*, 5(1):13–32.

Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. (1987). Hybrid Monte Carlo. *Physics letters B*, 195(2):216–222.

Embrechts, P., McNeil, A., and Straumann, D. (2002). Correlation and Dependence in Risk Management: Properties and Pitfalls. *Risk management: value at risk and beyond*, 1:176–223.

Engle, R. (2002). Dynamic Conditional Correlation: A Simple Class of Multivariate Generalized Autoregressive Conditional Heteroskedasticity Models. *Journal of Business & Economic Statistics*, 20(3):339–350.

Engle, R. and Figlewski, S. (2015). Modeling the Dynamics of Correlations Among Implied Volatilities. *Review of Finance*, 19(3):991–1018.

Engle, R. F. (1982). Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. *Econometrica: Journal of the Econometric Society*, 50:987–1007.

Engle, R. F. and Gallo, G. M. (2006). A Multiple Indicators Model for Volatility Using Intra-daily Data. *Journal of Econometrics*, 131(1-2):3–27.

Eraker, B. (2001). Mcmc Analysis of Diffusion Models with Application to Finance. *Journal of Business & Economic Statistics*, 19(2):177–191.

Evensen, G. (1994). Sequential Data Assimilation with A Nonlinear Quasi-geostrophic Model Using Monte Carlo Methods to Forecast Error Statistics. *Journal of Geophysical Research: Oceans*, 99(C5):10143–10162.

Figlewski, S. and Wang, X. (2000). Is the 'Leverage Effect' A Leverage Effect? Technical report, New York University - Stern School of Business.

Fleming, J. and Kirby, C. (2003). A Closer Look at the Relation Between GARCH and Stochastic Autoregressive Volatility. *Journal of financial econometrics*, 1(3):365–419.

Fomby, T. B., Johnson, S. R., and Hill, R. C. (1984). *Feasible Generalized Least Squares Estimation*, pages 147–169. Springer New York, New York, NY.

Fouque, J.-P., Papanicolaou, G., and Sircar, K. R. (2000). Mean-reverting Stochastic Volatility. *International Journal of theoretical and applied finance*, 3(01):101–142.

Fu, M. C., Laprise, S. B., Madan, D. B., Su, Y., and Wu, R. (2001). Pricing American Options: A Comparison of Monte Carlo Simulation Approaches. *Journal of Computational Finance*, 4(3):39–88.

Ghalanos, A. (2018). *Rugarch: Univariate GARCH Models*. R package version 1.4-0.

Golightly, A. and Wilkinson, D. J. (2006). Bayesian Sequential Inference for Nonlinear Multivariate Diffusions. *Statistics and Computing*, 16(4):323–338.

Gordon, N. J., Salmond, D. J., and Smith, A. F. (1993). Novel Approach to Nonlinear/non-Gaussian Bayesian State Estimation. 140:107–113.

Haas, M., Mittnik, S., and Paolella, M. S. (2004). Mixed Normal Conditional Heteroskedasticity. *Journal of Financial Econometrics*, 2(2):211–250.

Han, X. and Li, X. (2008). An Evaluation of the Nonlinear/non-Gaussian Filters for the Sequential Data Assimilation. *Remote Sensing of Environment*, 112(4):1434–1449.

Hansen, P. and Lunde, A. (2011). Forecasting Volatility Using High Frequency Data. In *The Oxford Handbook of Economic Forecasting*, pages 525–556. Blackwell: Oxford.

Hartigan, J. (1983). Asymptotic Normality of Posterior Distributions. In *Bayes Theory*, chapter 11, pages 107–118. Springer Series in Statistics.

Harvey, A., Ruiz, E., and Shephard, N. (1994). Multivariate Stochastic Variance Models. *The Review of Economic Studies*, 61(2):247–264.

Harvey, A. C. and Shephard, N. (1996). Estimation of An Asymmetric Stochastic Volatility Model for Asset Returns. *Journal of Business & Economic Statistics*, 14(4):429–434.

Herskovic, B., Kelly, B., Lustig, H., and Van Nieuwerburgh, S. (2016). The Common Factor in Idiosyncratic Volatility: Quantitative Asset Pricing Implications. *Journal of Financial Economics*, 119(2):249–283.

Heston, S. L. (1993). A Closed-form Solution for Options with Stochastic Volatility with Applications to Bond and Currency Options. *Review of financial studies*, 6(2):327–343.

Heston, S. L. (1997). A Simple New Formula for Options with Stochastic Volatility.

Heston, S. L. and Nandi, S. (2000). A Closed-form GARCH Option Valuation Model. *Review of Financial Studies*, 13(3):585–625.

Hoteit, I., Luo, X., and Pham, D.-T. (2012). Particle Kalman Filtering: A Nonlinear Bayesian Framework for Ensemble Kalman Filters. *Monthly weather review*, 140(2):528–542.

Hull, J. and White, A. (1987). The Pricing of Options on Assets with Stochastic Volatilities. *The journal of finance*, 42(2):281–300.

Hurn, A. S., Lindsay, K. A., and McClelland, A. J. (2015). Estimating the Parameters of Stochastic Volatility Models Using Option Price Data. *Journal of Business & Economic Statistics*, 33(4):579–594.

Ibanez, A. and Zapatero, F. (2004). Monte Carlo Valuation of American Options Through Computation of the Optimal Exercise Frontier. *Journal of Financial and Quantitative Analysis*, 39(2):253–275.

Ishihara, T. and Omori, Y. (2012). Efficient Bayesian Estimation of A Multivariate Stochastic Volatility Model with Cross Leverage and Heavy-tailed Errors. *Computational Statistics & Data Analysis*, 56(11):3674–3689.

Jacquier, E., Polson, N. G., and Rossi, P. (1994). Bayesian Analysis of Stochastic Volatility Models. *Journal of Business & Economic Statistics*, 12(4):371–89.

Jacquier, E., Polson, N. G., and Rossi, P. E. (2004). Bayesian Analysis of Stochastic Volatility Models with Fat-tails and Correlated Errors. *Journal of Econometrics*, 122(1):185–212.

Jacquier, E., Polson, N. G., Rossi, P. E., et al. (1995). Models and Priors for Multivariate Stochastic Volatility. Technical report, CIRANO.

Jacquier, E., Polson, N. G., Rossi, P. E., et al. (1999). Stochastic Volatility: Univariate and Multivariate Extensions. Technical report, CIRANO.

Jiang, G. J. and Tian, Y. S. (2007). Extracting Model-free Volatility From Option Prices: An Examination of the VIX Index. *The Journal of Derivatives*, 14(3):35–60.

Julier, S. J. and Uhlmann, J. K. (1997). New Extension of the Kalman Filter to Nonlinear Systems. In *AeroSense'97*, pages 182–193. International Society for Optics and Photonics.

Kalman, R. E. et al. (1960). A New Approach to Linear Filtering and Prediction Problems. *Journal of basic Engineering*, 82(1):35–45.

Kandepu, R., Foss, B., and Imsland, L. (2008). Applying the Unscented Kalman Filter for Nonlinear State Estimation. *Journal of process control*, 18(7-8):753–768.

Kastner, G., Frühwirth-Schnatter, S., and Lopes, H. F. (2017). Efficient Bayesian Inference for Multivariate Factor Stochastic Volatility Models. *Journal of Computational and Graphical Statistics*, (just-accepted).

Katzfuss, M., Stroud, J. R., and Wikle, C. K. (2016). Understanding the Ensemble Kalman Filter. *The American Statistician*, 70(4):350–357.

Kim, S., Shephard, N., and Chib, S. (1998). Stochastic Volatility: Likelihood Inference and Comparison with ARCH Models. *The Review of Economic Studies*, 65(3):361–393.

Kou, S., Olding, B. P., Lysy, M., and Liu, J. S. (2012). A Multiresolution Method for Parameter Estimation of Diffusion Processes. *Journal of the American Statistical Association*, 107(500):1558–1574.

Kraus, A. and Litzenberger, R. H. (1976). Skewness Preference and the Valuation of Risk Assets. *The Journal of Finance*, 31(4):1085–1100.

Kristensen, K. (2017). *tmbstan: MCMC Sampling From 'TMB' Model Object Using 'Stan'*. R package version 1.0.0.

Kristensen, K., Nielsen, A., Berg, C. W., Skaug, H., and Bell, B. M. (2016). TMB: Automatic Differentiation and Laplace Approximation. *Journal of Statistical Software*, 70(5):1–21.

Laplace, P.-S. (1774). Memoir on the Probability of the Causes of Events. *Stat. Sci*, 1(3):364–378.

LaViola, J. J. (2003). A Comparison of Unscented and Extended Kalman Filtering for Estimating Quaternion Motion. 3:2435–2440.

Ledoit, O. and Santa-Clara, P. (1998). Relative Pricing of Options with Stochastic Volatility. *University of California-Los Angeles Finance Working Paper*.

Ledoit, O., Santa-Clara, P., and Yan, S. (2002). Relative Pricing of Options with Stochastic Volatility. Technical report, Anderson Graduate School of Management, University of California, Los Angeles.

Lewis, A. L. (2009). *Option Valuation Under Stochastic Volatility II*. Finance Press: Newport Beach, CA.

Liesenfeld, R. and Richard, J.-F. (2003). Univariate and Multivariate Stochastic Volatility Models: Estimation and Diagnostics. *Journal of empirical finance*, 10(4):505–531.

Liu, J. S. and Chen, R. (1998). Sequential Monte Carlo Methods for Dynamic Systems. *Journal of the American statistical association*, 93(443):1032–1044.

Longstaff, F. A. and Schwartz, E. S. (2001). Valuing American Options by Simulation: A Simple Least-squares Approach. *The review of financial studies*, 14(1):113–147.

Lopes, H., McCulloch, R., and Tsay, R. (2012). Cholesky Stochastic Volatility Models for High-dimensional Time Series. *Discussion papers*.

Lysy, M. and Tong, J. (2017). *msde: Bayesian Inference for Multivariate Stochastic Differential Equations*. R package version 1.0.2.

Malik, S. and Pitt, M. K. (2011). Particle Filters for Continuous Likelihood Evaluation and Maximisation. *Journal of Econometrics*, 165(2):190–209.

Mandelbrot, B. B. (1963). The Variation of Certain Speculative Prices. 36:394–419.

Mardia, K. V. and Kent, J. (1979). Bibby. JM Multivariate analysis. *London: Academic*.

Martens, M. and Van Dijk, D. (2007). Measuring Volatility with the Realized Range. *Journal of Econometrics*, 138(1):181–207.

Maruyama, G. (1955). Continuous Markov Processes and Stochastic Equations. *Rendiconti del Circolo Matematico di Palermo*, 4(1):48–90.

McAleer, M. and Medeiros, M. C. (2008). Realized Volatility: A Review. *Econometric Reviews*, 27(1-3):10–45.

Mil'shtein, G. (1979). A Method of Second-order Accuracy Integration of Stochastic Differential Equations. *Theory of Probability & Its Applications*, 23(2):396–401.

Mil'shtejn, G. (1975). Approximate Integration of Stochastic Differential Equations. *Theory of Probability & Its Applications*, 19(3):557–562.

Nelson, D. B. (1990). Arch Models As Diffusion Approximations. *Journal of econometrics*, 45(1):7–38.

Patton, A. J. (2011). Volatility Forecast Comparison Using Imperfect Volatility Proxies. *Journal of Econometrics*, 160(1):246–256.

Pedersen, A. R. (1995). A New Approach to Maximum Likelihood Estimation for Stochastic Differential Equations Based on Discrete Observations. *Scandinavian journal of statistics*, pages 55–71.

Percy, D. F. (1992). Prediction for Seemingly Unrelated Regressions. *Journal of the Royal Statistical Society. Series B (Methodological)*, 54:243–252.

Perelló, J., Masoliver, J., and Anento, N. (2004). A Comparison Between Several Correlated Stochastic Volatility Models. *Physica A: Statistical Mechanics and its Applications*, 344(1):134–137.

Perelló, J., Sircar, R., and Masoliver, J. (2008). Option Pricing Under Stochastic Volatility: the Exponential Ornstein–Uhlenbeck Model. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(06):P06010.

Philipov, A. and Glickman, M. E. (2006). Factor Multivariate Stochastic Volatility Via Wishart Processes. *Econometric Reviews*, 25(2-3):311–334.

Phillips, P. C. and Yu, J. (2009). Maximum Likelihood and Gaussian Estimation of Continuous Time Models in Finance. In *Handbook of financial time series*. Springer.

Pitt, M. K., Malik, S., and Doucet, A. (2014). Simulated Likelihood Inference for Stochastic Volatility Models Using Continuous Particle Filtering. *Annals of the Institute of Statistical Mathematics*, 66(3):527–552.

Plett, G. L. (2006). Sigma-point Kalman Filtering for Battery Management Systems of LiPB-based HEV Battery Packs: Part 2: Simultaneous State and Parameter Estimation. *Journal of power sources*, 161(2):1369–1384.

Polson, N. G. and Stroud, J. R. (2003). Bayesian Inference for Derivative Prices.

Price, H. M., Ozawa, T., and Goldman, N. (2017). Synthetic Dimensions for Cold Atoms From Shaking A Harmonic Trap. *Physical Review A*, 95(2):023607.

Rogers, L. C. (2002). Monte Carlo Valuation of American Options. *Mathematical Finance*, 12(3):271–286.

Rogers, L. C. G. and Satchell, S. E. (1991). Estimating Variance From High, Low and Closing Prices. *The Annals of Applied Probability*, 1:504–512.

Rosenblatt, M. (1952). Remarks on a Multivariate Transformation. *The annals of mathematical statistics*, 23(3):470–472.

Sarkka, S. (2007). On Unscented Kalman Filtering for State Estimation of Continuous-time Nonlinear Systems. *IEEE Transactions on automatic control*, 52(9):1631–1641.

Särkkä, S., Hartikainen, J., Mbalawata, I. S., and Haario, H. (2015). Posterior Inference on Parameters of Stochastic Differential Equations Via Non-linear Gaussian Filtering and Adaptive MCMC. *Statistics and Computing*, 25(2):427–437.

Scott, L. O. (1987). Option Pricing When the Variance Changes Randomly: Theory, Estimation, and An Application. *Journal of Financial and Quantitative analysis*, 22(04):419–438.

Silverman, B. W. (2018). *Density Estimation for Statistics and Data Analysis*. Routledge.

Snyder, C., Bengtsson, T., Bickel, P., and Anderson, J. (2008). Obstacles to High-dimensional Particle Filtering. *Monthly Weather Review*, 136(12):4629–4640.

Song, P. X.-K. (2000). Monte Carlo Kalman Filter and Smoothing for Multivariate Discrete State Space Models. *Canadian Journal of Statistics*, 28(3):641–652.

Srivastava, V. and Dwivedi, T. (1979). Estimation of Seemingly Unrelated Regression Equations: A Brief Survey. *Journal of Econometrics*, 10(1):15–32.

Stordal, A. S., Karlsen, H. A., Nævdal, G., Skaug, H. J., and Vallès, B. (2011). Bridging the Ensemble Kalman Filter and Particle Filters: the Adaptive Gaussian Mixture Filter. *Computational Geosciences*, 15(2):293–305.

Szimayer, A., Dimitroff, G., and Lorenz, S. (2009). *A Parsimonious Multi-asset Heston Model: Calibration and Derivative Pricing*. Fraunhofer-Institut für Techno-und Wirtschaftsmathematik, Fraunhofer (ITWM).

Tilley, J. A. (1993). Valuing American Options in A Path Simulation Model. *Transactions of the Society of Actuaries*, 45:83–104.

Uhlig, H. (1997). Bayesian Vector Autoregressions with Stochastic Volatility. *Econometrica: Journal of the Econometric Society*, 65:59–73.

Van Der Merwe, R., Wan, E., and Julier, S. (2004a). Sigma-point Kalman Filters for Nonlinear Estimation and Sensor-fusion: Applications to Integrated Navigation. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*.

Van Der Merwe, R., Wan, E. A., et al. (2004b). Sigma-point Kalman Filters for Integrated Navigation. In *Proceedings of the 60th Annual Meeting of the Institute of Navigation (ION)*.

Visser, M. P. (2008). Forecasting S&P 500 Daily Volatility Using A Proxy for Downward Price Pressure. Technical report, Korteweg-de Vries Instute for Mathematics, University of Amsterdam.

Wan, E. A. and Van Der Merwe, R. (2000). The unscented Kalman Filter for Nonlinear Estimation. In *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*. Ieee.

Weerts, A. H. and El Serafy, G. Y. (2006). Particle Filtering and Ensemble Kalman Filtering for State Updating with Hydrological Conceptual Rainfall-runoff Models. *Water resources research*, 42(9).

Whaley, R. E. (1993). Derivatives on Market Volatility: Hedging Tools Long Overdue. *The journal of Derivatives*, 1(1):71–84.

Whaley, R. E. (2009). Understanding the VIX. *The Journal of Portfolio Management*, 35(3):98–105.

Wiggins, J. B. (1987). Option Values Under Stochastic Volatility: Theory and Empirical Estimates. *Journal of financial economics*, 19(2):351–372.

Wood, S. N. (2010). Statistical Inference for Noisy Nonlinear Ecological Dynamic Systems. *Nature*, 466(7310):1102.

Xiong, K., Zhang, H., and Chan, C. (2006). Performance Evaluation of UKF-based Nonlinear Filtering. *Automatica*, 42(2):261–270.

Yu, J. (2005). On Leverage in A Stochastic Volatility Model. *Journal of Econometrics*, 127(2):165–178.

Yu, J. and Meyer, R. (2006). Multivariate Stochastic Volatility Models: Bayesian Estimation and Model Comparison. *Econometric Reviews*, 25(2-3):361–384.

Zellner, A. (1962). An Efficient Method of Estimating Seemingly Unrelated Regressions and Tests for Aggregation Bias. *Journal of the American statistical Association*, 57(298):348–368.

Zellner, A. and Ando, T. (2010). A Direct Monte Carlo Approach for Bayesian Analysis of the Seemingly Unrelated Regression Model. *Journal of Econometrics*, 159(1):33–45.

APPENDICES

Appendix A

Gibbs Sampler for the RML Model

Let us start by defining some notations.

$$\boldsymbol{Y} = \Big[y_{ti}\Big]_{t=1,\ldots,n}^{i=1,\ldots,q} = \begin{pmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_n \end{pmatrix} = \big(\boldsymbol{y}_{(1)}, \cdots, \boldsymbol{y}_{(q)}\big), \qquad \boldsymbol{y}_t = (y_{t1}, \ldots, y_{tq}), \qquad \boldsymbol{y}_{(i)} = \begin{pmatrix} y_{1i} \\ \vdots \\ y_{ni} \end{pmatrix}$$

$$\boldsymbol{X}_i = \Big[x_{tj}^{(i)}\Big]_{t=1,\ldots,n}^{j=1,\ldots,p_i} = \begin{pmatrix} \boldsymbol{x}'_{1i} \\ \vdots \\ \boldsymbol{x}'_{ni} \end{pmatrix}, \qquad \boldsymbol{X} = (\boldsymbol{X}_1, \ldots, \boldsymbol{X}_q).$$

As we can see, we define $\boldsymbol{Y}$ both by column and by row, and we define $\boldsymbol{X}$ by column-block and by row. We use a bold capital letter to define matrix, a bold lower-case letter to define vector and a regular lower-case letter to define scalar. Also

- n is the sample size
- q is the number of equations
- p_i, $i = 1, \cdots, q$ are the number of parameters in corresponding equations
- p is the maximum of p_i, $i = 1, \cdots, q$
- $d = \sum_i p_i$ is the total number of β coefficient.

There exists a conjugate prior for $\boldsymbol{\theta} = (\boldsymbol{\beta}, \boldsymbol{\Sigma})$ as

$$\begin{aligned}\boldsymbol{\Sigma} &\sim \mathcal{W}_q^{-1}(\boldsymbol{\Psi}, \nu),\\ \boldsymbol{\beta}|\boldsymbol{\Sigma} &\sim \mathcal{N}_d(\boldsymbol{\lambda}, \boldsymbol{\Omega}).\end{aligned}$$

Notice the log-density of the prior are

$$\begin{aligned}\ell(\boldsymbol{\Sigma}) &= -\frac{\nu+q+1}{2}\log(|\boldsymbol{\Sigma}|) - \frac{1}{2}\mathrm{Tr}(\boldsymbol{\Psi}\boldsymbol{\Sigma}^{-1}),\\ \ell(\boldsymbol{\beta}) &= -\frac{1}{2}\log(|\boldsymbol{\Omega}|) - \frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\lambda})'\boldsymbol{\Omega}^{-1}(\boldsymbol{\beta}-\boldsymbol{\lambda}).\end{aligned}$$

Together with the model log density of

$$\ell(\boldsymbol{Y}, \boldsymbol{X}|\boldsymbol{\theta}) = -\frac{n}{2}\log(|\boldsymbol{\Sigma}|) - \frac{1}{2}\sum_{t=1}^{n}\big(\boldsymbol{y}_t - \boldsymbol{w}_t\big)'\boldsymbol{\Sigma}^{-1}\big(\boldsymbol{y}_t - \boldsymbol{w}_t\big),$$

where $\boldsymbol{w}_t = \big(\boldsymbol{x}_{t1}'\boldsymbol{\beta}_1, \ldots, \boldsymbol{x}_{tq}'\boldsymbol{\beta}_q\big)$, we can derive the posterior density. Although the prior and the model densities are both straightforward, but the posterior density of $\boldsymbol{\theta}$ does not belong to any well-known distributions. However, the conditional posterior distribution of $\boldsymbol{\Sigma}|\boldsymbol{\beta}, \boldsymbol{Y}, \boldsymbol{X}$ follows an inverse Wishart distribution and $\boldsymbol{\beta}|\boldsymbol{\Sigma}, \boldsymbol{Y}, \boldsymbol{X}$ follows a Gaussian distribution. In other words, a Gibbs sampler for the posterior parameters can be found.

A.1 Sigma Update

The posterior distribution of $\boldsymbol{\Sigma}|\boldsymbol{\beta}, \boldsymbol{Y}, \boldsymbol{X}$ is

$$p(\boldsymbol{\Sigma}|\boldsymbol{\beta}, \boldsymbol{Y}, \boldsymbol{X}) = -\frac{\nu+q+n+1}{2}\log(|\boldsymbol{\Sigma}|) - \frac{1}{2}\mathrm{Tr}\left(\boldsymbol{\Psi}\boldsymbol{\Sigma}^{-1} + \sum_{t=1}^{n}\Big(\boldsymbol{y}_t - \boldsymbol{w}_t\Big)\boldsymbol{\Sigma}^{-1}\Big(\boldsymbol{y}_t - \boldsymbol{w}_t\Big)'\right).$$

With the property of the matrix trace, we have

$$\mathrm{Tr}\left(\boldsymbol{\Psi}\boldsymbol{\Sigma}^{-1} + \sum_{t=1}^{n}\Big(\boldsymbol{y}_t - \boldsymbol{w}_t\Big)\boldsymbol{\Sigma}^{-1}\Big(\boldsymbol{y}_t - \boldsymbol{w}_t\Big)'\right) = \mathrm{Tr}\left(\Big(\boldsymbol{\Psi} + \sum_{t=1}^{n}\big(\boldsymbol{y}_t - \boldsymbol{w}_t\big)'\big(\boldsymbol{y}_t - \boldsymbol{w}_t\big)\Big)\boldsymbol{\Sigma}^{-1}\right)$$

which means that

$$\boldsymbol{\Sigma}|\boldsymbol{\beta}, \boldsymbol{Y}, \boldsymbol{X} \sim \mathcal{W}_q^{-1}(\boldsymbol{\Psi} + \boldsymbol{D}, \nu + n)$$

where $\boldsymbol{D} = \sum_{t=1}^{n} \big(\boldsymbol{y}_t - \boldsymbol{w}_t\big)'\big(\boldsymbol{y}_t - \boldsymbol{w}_t\big)$, which is a $q \times q$ matrix.

Notice that $\boldsymbol{D}$ is in quite a complicated form, we find we can rewrite $\boldsymbol{D}$ in the following simpler form. We inspect the (i, j) element of $\boldsymbol{D}$ first,

$$\begin{aligned} D_{ij} &= \sum_{t=1}^{n} \Big[\big(\boldsymbol{y}_t - \boldsymbol{w}_t\big)'\big(\boldsymbol{y}_t - \boldsymbol{w}_t\big)\Big]_{ij} = \sum_{t=1}^{n} (y_{ti} - \boldsymbol{x}'_{ti}\boldsymbol{\beta}_i)(y_{tj} - \boldsymbol{x}'_{tj}\boldsymbol{\beta}_j) \\ &= \big(\boldsymbol{y}_{(i)} - \boldsymbol{X}_i\boldsymbol{\beta}_i\big)'\big(\boldsymbol{y}_{(j)} - \boldsymbol{X}_j\boldsymbol{\beta}_j\big) \\ &= \big(\boldsymbol{y}'_{(i)}\boldsymbol{y}_{(j)}\big) - \boldsymbol{\beta}'_i\big(\boldsymbol{X}'_i\boldsymbol{y}_{(j)}\big) - \big(\boldsymbol{y}'_{(i)}\boldsymbol{X}_j\big)\boldsymbol{\beta}_j + \boldsymbol{\beta}'_i\big(\boldsymbol{X}'_i\boldsymbol{X}_j\big)\boldsymbol{\beta}_j \end{aligned}$$

Apparently, $\boldsymbol{y}'_{(i)}\boldsymbol{y}_{(j)}$, $\boldsymbol{X}'_i\boldsymbol{y}_{(j)}$, $\boldsymbol{y}'_{(i)}\boldsymbol{X}_j$ and $\boldsymbol{X}'_i\boldsymbol{X}_j$ can be found as blocks of $\boldsymbol{X}'\boldsymbol{X}$, $\boldsymbol{Y}'\boldsymbol{X}$ and $\boldsymbol{Y}'\boldsymbol{Y}$, which can be calculate only once in advance. We do not need to recalculate those sufficient statistics again in every Gibbs update.

In summary, we have procedures as

1. Calculate $\boldsymbol{D}$. This step costs $\mathcal{O}(p^2q^2)$
2. Calculate L_q, the lower Cholesky decomposition of $\boldsymbol{D} + \Psi$. This step costs $\mathcal{O}(q^3)$
3. Simulate Φ such that $\Phi\Phi' \sim \mathcal{W}(I, n + \nu)$ by q χ^2 and $q(q-1)/2$ normal, cost $\mathcal{O}(q^2)$
4. Calculate $(\boldsymbol{D}+\boldsymbol{\Psi})^{-1/2}$ times Φ by a lower triangular solve algorithm to obtain $\boldsymbol{\Sigma}^{-1/2}$. This step costs $\mathcal{O}(q^3)$
5. Cross-product above matrix, which costs $\mathcal{O}(q^3)$

In the end, we have $\boldsymbol{\Sigma}^{-1}$, which can be shown in the following algorithm is the only thing needed for updating $\boldsymbol{\beta}$. Eventually the algorithm will calculate $\boldsymbol{\Sigma}$ before final output, which costs $\mathcal{O}(q^3)$.

A.2 Beta Update

The posterior distribution of $\boldsymbol{\beta}|\boldsymbol{\Sigma}, \boldsymbol{Y}, \boldsymbol{X}$ is

$$p(\boldsymbol{\beta}|\boldsymbol{\Sigma}, \boldsymbol{Y}, \boldsymbol{X}) = -\frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\lambda})'\boldsymbol{\Omega}^{-1}(\boldsymbol{\beta}-\boldsymbol{\lambda}) - \frac{1}{2}\sum_{t=1}^{n}\big(\boldsymbol{y}_t - \boldsymbol{w}_t\big)\boldsymbol{\Sigma}^{-1}\big(\boldsymbol{y}_t - \boldsymbol{w}_t\big)'.$$

In the later half of the right hand side of the above equation, $\boldsymbol{\beta}$ is obscured. In order to show, first we rewrite $\boldsymbol{w}_t$

$$\boldsymbol{w}_t = \big(\boldsymbol{x}'_{t1}\boldsymbol{\beta}_1, \ldots, \boldsymbol{x}'_{tq}\boldsymbol{\beta}_q\big) = \boldsymbol{\beta}' \begin{pmatrix} \boldsymbol{x}_{t1} & 0 & \cdots & 0 \\ 0 & \boldsymbol{x}_{t2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{x}_{tq} \end{pmatrix}_{d\times q}.$$

Now our objective is to rewrite $\sum_{t=1}^{n}\big(\boldsymbol{y}_t-\boldsymbol{w}_t\big)\boldsymbol{\Sigma}^{-1}\big(\boldsymbol{y}_t-\boldsymbol{w}_t\big)'$ into a form of $\big(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}\big)'\boldsymbol{V}^{-1}\big(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}\big)$ by solving $\hat{\boldsymbol{\beta}}$ and $\boldsymbol{V}$ as functions of $\boldsymbol{\Sigma}$, $\boldsymbol{Y}$ and $\boldsymbol{X}$. We can derive

$$\sum_{t=1}^{n}\big(\boldsymbol{y}_t - \boldsymbol{w}_t\big)\boldsymbol{\Sigma}^{-1}\big(\boldsymbol{y}_t - \boldsymbol{w}_t\big)' = \sum_{t=1}^{n}\big(-2\boldsymbol{w}_t\boldsymbol{\Sigma}^{-1}\boldsymbol{y}'_t + \boldsymbol{w}_t\boldsymbol{\Sigma}^{-1}\boldsymbol{w}'_t\big) + const$$

Notice that the term $\sum_t \boldsymbol{w}_t\boldsymbol{\Sigma}^{-1}\boldsymbol{w}'_t$ determines $\boldsymbol{V}$ since it is the only term involving both $\boldsymbol{\beta}$ and $\boldsymbol{\beta}'$. Together with the rewrite of $\boldsymbol{w}_t$ above, it is easy to show

$$\boldsymbol{V}^{-1} = \sum_{t=1}^{n}\left[\begin{pmatrix} \boldsymbol{x}_{t1} & 0 & \cdots & 0 \\ 0 & \boldsymbol{x}_{t2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{x}_{tq} \end{pmatrix}\boldsymbol{\Sigma}^{-1}\begin{pmatrix} \boldsymbol{x}'_{t1} & 0 & \cdots & 0 \\ 0 & \boldsymbol{x}'_{t2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{x}'_{tq} \end{pmatrix}\right]$$

Define $\boldsymbol{\Sigma}^{-1} = [\tau_{i,j}]_{j=1,\cdots,q}^{i=1,\cdots,q}$, then we can inspect the (i,j) block (it is not a scalar) of $\boldsymbol{V}^{-1}$,

$$\big[\boldsymbol{V}^{-1}\big]_{ij} = \sum_{t=1}^{n}\boldsymbol{x}_{ti}\tau_{i,j}\boldsymbol{x}'_{tj} = \tau_{i,j}\sum_{t=1}^{n}\boldsymbol{x}_{ti}\boldsymbol{x}'_{tj} = \tau_{i,j}\boldsymbol{X}'_i\boldsymbol{X}_j$$

which has concluded the structure of $\boldsymbol{V}$. Also for the similar reason, we can determine $\hat{\boldsymbol{\beta}}$ by $2\boldsymbol{w}_t\boldsymbol{\Sigma}^{-1}\boldsymbol{y}_t'$. Since

$$\boldsymbol{w}_t\boldsymbol{\Sigma}^{-1}\boldsymbol{y}_t' = \boldsymbol{\beta}'\sum_{t=1}^{n}\left[\begin{pmatrix}\boldsymbol{x}_{t1} & 0 & \cdots & 0\\ 0 & \boldsymbol{x}_{t2} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \boldsymbol{x}_{tq}\end{pmatrix}\boldsymbol{\Sigma}^{-1}\boldsymbol{y}_t'\right] = \boldsymbol{\beta}'\boldsymbol{V}^{-1}\hat{\boldsymbol{\beta}},$$

also notice the ith row vector of $\boldsymbol{V}^{-1}\hat{\boldsymbol{\beta}}$, (a d-dimensional vector) as

$$\left[\boldsymbol{V}^{-1}\hat{\boldsymbol{\beta}}\right]_i = \sum_{t=1}^{n}\boldsymbol{x}_{ti}(\tau_{1,j},\cdots,\tau_{q,j})\boldsymbol{y}_t' = \sum_{t=1}^{n}\sum_{j=1}^{q}\boldsymbol{x}_{ti}\tau_{i,j}y_{tj} = \sum_{j=1}^{q}\tau_{i,j}\sum_{t=1}^{n}\boldsymbol{x}_{ti}y_{tj} = \sum_{j=1}^{q}\tau_{i,j}\boldsymbol{X}_i'\boldsymbol{y}_{(j)},$$

which concludes the structure of $\boldsymbol{V}^{-1}\hat{\boldsymbol{\beta}}$ and later can be used to derive $\hat{\boldsymbol{\beta}}$.

Now let us summarize the above information to have a clear form of the sufficient statistics $\boldsymbol{V}$ and $\hat{\boldsymbol{\beta}}$. Define the notation $[\alpha]_{a,b}$ represents a $a\times b$ matrix with every element being α, then let us further define

$$\boldsymbol{\Sigma}_1 = \begin{array}{c} \\ p_1\\ \vdots\\ p_q\end{array}\begin{array}{c}\begin{array}{ccc} p_1 & \cdots & p_q\end{array}\\ \begin{pmatrix}[\tau_{11}]_{p_1,p_1} & \cdots & [\tau_{1q}]_{p_1,p_q}\\ \vdots & \ddots & \vdots\\ [\tau_{q1}]_{p_q,p_1} & \cdots & [\tau_{qq}]_{p_q,p_q}\end{pmatrix}\end{array}, \qquad \boldsymbol{\Sigma}_2 = \begin{array}{c} \\ p_1\\ \vdots\\ p_q\end{array}\begin{array}{c}\begin{array}{ccc} 1 & \cdots & 1\end{array}\\ \begin{pmatrix}[\tau_{11}]_{p_1,1} & \cdots & [\tau_{1q}]_{p_1,1}\\ \vdots & \ddots & \vdots\\ [\tau_{q1}]_{p_q,1} & \cdots & [\tau_{qq}]_{p_q,1}\end{pmatrix}\end{array},$$

then the sufficient statistics are

$$\boldsymbol{V}^{-1} = \boldsymbol{\Sigma}_1\circ\boldsymbol{X}'\boldsymbol{X}, \qquad \hat{\boldsymbol{\beta}} = \boldsymbol{V}\left(\boldsymbol{\Sigma}_2\circ\boldsymbol{X}'\boldsymbol{Y}\right)\begin{pmatrix}1\\ \vdots\\ 1\end{pmatrix}.$$

Now, we lead back to the posterior log-density of $\boldsymbol{\beta}|\boldsymbol{\Sigma},\boldsymbol{Y},\boldsymbol{X}$ as

$$\begin{aligned}\ell(\boldsymbol{\beta}|\boldsymbol{\Sigma},\boldsymbol{Y},\boldsymbol{X}) &= -\tfrac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\lambda})'\boldsymbol{\Omega}^{-1}(\boldsymbol{\beta}-\boldsymbol{\lambda}) - \tfrac{1}{2}(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}})'\boldsymbol{V}^{-1}(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}})\\ &= -\tfrac{1}{2}\Big(\boldsymbol{\beta}'(\boldsymbol{\Omega}^{-1}+\boldsymbol{V}^{-1})\boldsymbol{\beta} - 2\boldsymbol{\lambda}'\boldsymbol{\Omega}^{-1}\boldsymbol{\beta} - 2\hat{\boldsymbol{\beta}}'V^{-1}\boldsymbol{\beta}\Big)\\ &= -\tfrac{1}{2}\Big(\boldsymbol{\beta}-\big(\boldsymbol{B}\boldsymbol{\lambda}+(\boldsymbol{I}-\boldsymbol{B})\hat{\boldsymbol{\beta}}\big)\Big)'(\boldsymbol{V}^{-1}+\boldsymbol{\Omega}^{-1})\Big(\boldsymbol{\beta}-\big(\boldsymbol{B}\boldsymbol{\lambda}+(\boldsymbol{I}-\boldsymbol{B})\hat{\boldsymbol{\beta}}\big)\Big)\end{aligned}$$

where $\boldsymbol{B} = (\boldsymbol{\Omega}^{-1} + \boldsymbol{V}^{-1})^{-1}\boldsymbol{\Omega}^{-1}$ and $\boldsymbol{I} - \boldsymbol{B} = (\boldsymbol{\Omega}^{-1} + \boldsymbol{V}^{-1})^{-1}\boldsymbol{V}^{-1}$. Hence

$$\boldsymbol{\beta}|\boldsymbol{\Sigma}, \boldsymbol{Y}, \boldsymbol{X} \sim \mathcal{N}(\boldsymbol{B}\boldsymbol{\lambda} + (I - \boldsymbol{B})\hat{\boldsymbol{\beta}}, (\boldsymbol{\Omega}^{-1} + \boldsymbol{V}^{-1})^{-1}).$$

In summary, we have the following procedures to sample $\boldsymbol{\beta}$ as

1. Calculate $\boldsymbol{V}^{-1}$ and $\boldsymbol{V}^{-1}\hat{\boldsymbol{\beta}}$ according to $\boldsymbol{\Sigma}^{-1}$ from previous sigma update. This step costs $\mathcal{O}(d^2)$

2. Notice that in the following formula

$$\begin{aligned}\boldsymbol{B}\boldsymbol{\lambda} + (I - \boldsymbol{B})\hat{\boldsymbol{\beta}} &= (\boldsymbol{\Omega}^{-1} + \boldsymbol{V}^{-1})^{-1}\boldsymbol{\Omega}^{-1}\lambda + (\boldsymbol{\Omega}^{-1} + \boldsymbol{V}^{-1})^{-1}\boldsymbol{V}^{-1}\hat{\boldsymbol{\beta}} \\ &= (\boldsymbol{\Omega}^{-1} + \boldsymbol{V}^{-1})^{-1}(\boldsymbol{\Omega}^{-1}\lambda + \boldsymbol{V}^{-1}\hat{\boldsymbol{\beta}}),\end{aligned}$$

   $\boldsymbol{\Omega}^{-1}\lambda$ can be pre-calculated. Calculate L_d as the lower cholesky decomposition of $(\boldsymbol{\Omega}^{-1}+\boldsymbol{V}^{-1})$. Notice $(\boldsymbol{\Omega}^{-1}+\boldsymbol{V}^{-1})^{-1} = L_d'^{-1}L_d^{-1}$, so we will solve two sets of equations to get $L_d'^{-1}L_d^{-1}(\boldsymbol{\Omega}^{-1}\lambda + \boldsymbol{V}^{-1}\hat{\boldsymbol{\beta}})$. This step costs $\mathcal{O}(d^3)$

3. Generate d standard normal R_d, cost $\mathcal{O}(d)$

4. Solve one set of equations to get $L_d'^{-1}R_d$, then add the mean $\boldsymbol{B}\boldsymbol{\lambda} + (I - \boldsymbol{B})\hat{\boldsymbol{\beta}}$. This step costs $\mathcal{O}(d^3)$

As we suggest before, we do not need to know $\boldsymbol{\Sigma}$, but only $\boldsymbol{\Sigma}^{-1}$. For the improper prior, we just substitute in the corresponding prior statistics to be 0 and the result can be followed simply.

Appendix B

Bayesian Inference for Multivariate Diffusions

In this appendix, we describe the MCMC algorithm on the augmented space of the latent volatilities and the model parameters. Firstly, we define a generic SDE model with latent variables as

$$\mathrm{d}\boldsymbol{Y}_t = \boldsymbol{\Psi}(\boldsymbol{Y}_t, \boldsymbol{\theta})\,\mathrm{d}t + \boldsymbol{\Phi}^{1/2}(\boldsymbol{Y}_t, \boldsymbol{\theta})\,\mathrm{d}\boldsymbol{B}_t$$

where $\boldsymbol{B}_t$ is a vector of independent Brownian motions. The model variable vector $\boldsymbol{Y}_t = (\boldsymbol{X}_t, \boldsymbol{Z}_t)$, where $\boldsymbol{X}_t$ are observed variable vector of size p and $\boldsymbol{Z}_t$ are latent variable vector of size q. The mSV models fall into this generic model framework.

Those mSV models we deal with do not have closed-form transition densities $p(\boldsymbol{Y}_t|\boldsymbol{Y}_{t-1}, \boldsymbol{\theta})$. Hence we adopt the Euler-Maruyama approximation method by operating on the discretized SDE model of

$$\Delta\boldsymbol{Y}_t = \boldsymbol{\Psi}(\boldsymbol{Y}_t, \boldsymbol{\theta})\Delta\mathrm{T} + \boldsymbol{\Phi}^{1/2}(\boldsymbol{Y}_t, \boldsymbol{\theta})\Delta\boldsymbol{B}_t, \tag{B.1}$$

where $\Delta\boldsymbol{B}_t \sim \mathcal{N}_{p+q}(\boldsymbol{0}, \Delta\mathrm{T}\boldsymbol{I}_{p+q})$, so that

$$\Delta\boldsymbol{Y}_t \sim \mathcal{N}_{p+q}(\Delta\mathrm{T}\boldsymbol{\mu}_t, \Delta\mathrm{T}\boldsymbol{\Sigma}_t),$$

where $\boldsymbol{\mu}_t = \boldsymbol{\Psi}(\boldsymbol{Y}_t, \boldsymbol{\theta})$ and $\boldsymbol{\Sigma}_t = \boldsymbol{\Phi}(\boldsymbol{Y}_t, \boldsymbol{\theta})$. This Gaussian density is an approximation to the analytical transition density. In this paper, we deal with observations in daily frequency, on which the Euler-Maruyama approximation of the model likelihood based is generally usable (Phillips and Yu, 2009). In the following, we use this Gaussian density as the transition density $p(\boldsymbol{Y}_t|\boldsymbol{Y}_{t-1}, \boldsymbol{\theta})$.

In a Bayesian framework, we can obtain the posterior density $p(\boldsymbol{\theta}|\boldsymbol{X}_{0:N})$ by sampling from the augmented space of the model parameters and the latent variables, i.e., $p(\boldsymbol{Z}_{0:N}, \boldsymbol{\theta}|\boldsymbol{X}_{0:N})$, the density of which can be obtained by

$$p(\boldsymbol{Z}_{0:N}, \boldsymbol{\theta}|\boldsymbol{X}_{0:N}) \propto \pi(\boldsymbol{\theta}) p(\boldsymbol{Y}_{0:N}|\boldsymbol{\theta}) = \pi(\boldsymbol{\theta}) \prod_{t=1}^{N} p(\boldsymbol{Y}_t|\boldsymbol{Y}_{t-1}\boldsymbol{\theta}),$$

where $\pi(\boldsymbol{\theta})$ is the prior. Thus, we implement a Gibbs sampler of updating $\boldsymbol{Z}_{0:n}$ and $\boldsymbol{\theta}$ by conditioning on each other iteratively.

B.1 Update Latent Variables

Firstly, we sample from $p(\boldsymbol{Z}_{0:N}|\boldsymbol{X}_{0:N}, \boldsymbol{\theta})$ by the algorithm of Eraker (2001) in the following.

1. For $t = 0$, notice that

   $$p(\boldsymbol{Z}_0|\boldsymbol{X}_0, \boldsymbol{Y}_{1:N}, \boldsymbol{\theta}) \propto p(\boldsymbol{Y}_0|\boldsymbol{Y}_{1:n}, \boldsymbol{\theta}),$$

   and according to the Markov property

   $$p(\boldsymbol{Y}_0|\boldsymbol{Y}_{1:N}, \boldsymbol{\theta}) \propto p(\boldsymbol{Y}_0|\boldsymbol{Y}_1, \boldsymbol{\theta}) = |\Delta \mathrm{T} \boldsymbol{\Sigma}_0|^{-\frac{1}{2}} \times \exp\left(\tfrac{1}{\Delta \mathrm{T}} (\Delta \boldsymbol{Y}_0 - \boldsymbol{\mu}_0) \boldsymbol{\Sigma}_0^{-1} (\Delta \boldsymbol{Y}_0 - \boldsymbol{\mu}_0)'\right)$$

   which is the first transition density. An analytic draw might not exist so that a vanilla Metropolis-Hasting (MH) algorithm is used to update $\boldsymbol{Z}_0$.

2. For $t = 1 : (N-1)$, due to the Markov property, we have

   $$p(\boldsymbol{Z}_t|\boldsymbol{X}_t, \boldsymbol{Y}_{-t}, \boldsymbol{\theta}) \propto p(\boldsymbol{Y}_t|\boldsymbol{Y}_{t-1}, \boldsymbol{Y}_{t+1}, \boldsymbol{\theta}),$$

where $\boldsymbol{Y}_{-t}$ represents $\boldsymbol{Y}_{0:N}$ without $\boldsymbol{Y}_t$ and

$$\log\big(p(\boldsymbol{Y}_t|\boldsymbol{Y}_{t-1},\boldsymbol{Y}_{t+1},\boldsymbol{\theta})\big) = -\tfrac{1}{2}\log(|\boldsymbol{\Sigma}_{t-1}|\times|\boldsymbol{\Sigma}_t|) - \\ \tfrac{1}{\Delta\mathrm{T}}(\Delta\boldsymbol{Y}_{t-1}-\boldsymbol{\mu}_{t-1})\boldsymbol{\Sigma}_{t-1}^{-1}(\Delta\boldsymbol{Y}_{t-1}-\boldsymbol{\mu}_{t-1})' - \tfrac{1}{\Delta\mathrm{T}}(\Delta\boldsymbol{Y}_t-\boldsymbol{\mu}_t)\boldsymbol{\Sigma}_t^{-1}(\Delta\boldsymbol{Y}_t-\boldsymbol{\mu}_t)'$$

where an analytic draw is usually not available. Hence, we use the MH algorithm on the ratio of target density $p(\boldsymbol{Z}_t|\boldsymbol{X}_t,\boldsymbol{Y}_{t-1},\boldsymbol{Y}_{t+1},\boldsymbol{\theta})$ and the proposal density $q(\boldsymbol{Z}_t|\boldsymbol{X}_t,\boldsymbol{Y}_{t-1},\boldsymbol{Y}_{t+1},\boldsymbol{\theta})$, which can be found in the following way. By Eraker (2001), an approximation to $p(\boldsymbol{Y}_t|\boldsymbol{Y}_{t-1},\boldsymbol{Y}_{t+1})$ is

$$\mathcal{N}_{p+q}\big(\tfrac{1}{2}(\boldsymbol{Y}_{t-1}+\boldsymbol{Y}_{t+1}),\tfrac{1}{2}\boldsymbol{\Sigma}_{t-1}\Delta\mathrm{T}\big).$$

We further define

$$\tfrac{1}{2}\boldsymbol{\Sigma}_{t-1}\Delta\mathrm{T} = \begin{matrix} \\ p \\ q \end{matrix}\begin{matrix} \begin{matrix} p & \quad q \end{matrix} \\ \begin{pmatrix} \boldsymbol{\Sigma}_{11,t} & \boldsymbol{\Sigma}_{12,t} \\ \boldsymbol{\Sigma}_{12,t}' & \boldsymbol{\Sigma}_{22,t} \end{pmatrix} \end{matrix}, \quad \tfrac{1}{2}(\boldsymbol{Y}_{t-1}+\boldsymbol{Y}_{t+1}) = \begin{matrix} p \\ q \end{matrix}\begin{pmatrix} \boldsymbol{m}_{X,t} \\ \boldsymbol{m}_{Z,t} \end{pmatrix}$$

then the proposal density $q(\boldsymbol{Z}_t|\boldsymbol{X}_t,\boldsymbol{Y}_{-t},\boldsymbol{\theta})$ follows the conditional Gaussian distribution

$$\mathcal{N}\big(\boldsymbol{m}_{Z,t}-\boldsymbol{\Sigma}_{12,t}'\boldsymbol{\Sigma}_{11,t}^{-1}\boldsymbol{m}_{X,t},\boldsymbol{\Sigma}_{22,t}-\boldsymbol{\Sigma}_{12,t}'\boldsymbol{\Sigma}_{11,t}^{-1}\boldsymbol{\Sigma}_{12,t}\big).$$

3. For $t=N$, due to that

$$p(\boldsymbol{Z}_N|\boldsymbol{X}_N,\boldsymbol{Y}_{-N},\boldsymbol{\theta}) \propto p(\boldsymbol{Y}_N|\boldsymbol{Y}_{N-1},\boldsymbol{\theta}),$$

the proposal density is

$$\mathcal{N}_{p+q}\big(\boldsymbol{Y}_{N-1},\tfrac{1}{2}\boldsymbol{\Sigma}_{N-1}\Delta\mathrm{T}\big)$$

and MH algorithm on the ratio of the target and proposal densities as the previous step can be applied.

B.2 Update Parameters

Conditioning on the latent variables $\boldsymbol{Z}_{0:N}$, we can evaluate the model likelihood conditioning on a specific set of the parameters $\boldsymbol{\theta}$. The posterior density of the parameters

conditional on the full data $(\boldsymbol{X}_{0:N}, \boldsymbol{Z}_{0:N})$ can be obtained as

$$p(\boldsymbol{\theta}|\boldsymbol{X}_{0:N}, \boldsymbol{Z}_{0:N}) = \pi(\boldsymbol{\theta}) \prod_{t=1}^{N} p(\boldsymbol{Y}_t|\boldsymbol{Y}_{t-1}\boldsymbol{\theta})$$

where $\pi(\boldsymbol{\theta})$ is the Bayesian prior. Normally, there are multiple parameters in $\boldsymbol{\theta}$, so we update them one by one with the vanilla MH algorithm in a Gibbs sampler framework.

B.3 Heston vs. eOU Model

Prior to applying the Euler-Maruyama discretization (B.1), it is often advisable to transform the SDE. For example, the eOU model of Scott (1987); Fouque et al. (2000) is usually discretized in the variables $X_t = \log S_t$ and $Z_t = \log V_t$, whence by Itō's lemma the SDE becomes

$$\begin{aligned} \mathrm{d}X_t &= (\alpha - \tfrac{1}{2}e^{2Z_t})\,\mathrm{d}t + e^{Z_t}\,\mathrm{d}B_t^{\mathsf{X}} \\ \mathrm{d}Z_t &= \gamma(\mu - Z_t)\,\mathrm{d}t + \sigma\,\mathrm{d}B_t^{\mathsf{Z}}, \qquad \mathrm{cor}(\mathrm{d}B_t^{\mathsf{X}}, \mathrm{d}B_t^{\mathsf{Z}}) = \rho. \end{aligned}$$

On the other hand, for Heston's model (Heston, 1993) the discretization is usually performed along the variables $X_t = \log S_t$ and $Z_t = V_t$ itself, such that the SDE model becomes

$$\begin{aligned} \mathrm{d}X_t &= (\alpha - \tfrac{1}{2}V_t^2)\,\mathrm{d}t + V_t\,\mathrm{d}B_t^{\mathsf{X}} \\ \mathrm{d}Z_t &= (\beta Z_t^{-1} - \tfrac{1}{2}\gamma Z_t)\,\mathrm{d}t + \tfrac{1}{2}\sigma\,\mathrm{d}B_t^{\mathsf{Z}}, \qquad \mathrm{cor}(\mathrm{d}B_t^{\mathsf{X}}, \mathrm{d}B_t^{\mathsf{Z}}) = \rho, \end{aligned}$$

where $\beta = \gamma\mu - \frac{1}{4}\sigma^2$. The reason for this particular change-of-variables transformation is that the volatility diffusion function becomes constant $(\sigma/2)$, in which case the Euler-Maruyama approximation has a faster rate of convergence (Mil'shtein, 1979). However, for the eOU model we have $X_t, Z_t \in \mathbb{R}$, whereas for Heston's model we have $Z_t > 0$, such that the Euler-Maruyama Gaussian approximation can be poor for certain parameter combinations (Andersen, 2007), resulting in slower MCMC convergence. It is primarily for this reason that we elected to use the eOU model instead of Heston's in this paper.

Appendix C

Unscented Kalman Filter

This Appendix adopts the unscented Kalman filter of Wan and Van Der Merwe (2000) into the generalized Gaussian state-space model (4.2).

Before we introduce the unscented Kalman filter, let us introduce the unscented transformation first. Suppose we have a random vector X, for which we know the mean and variance μ and Σ. The objective is to find out the mean and variance of $f(X)$, where f can be a nonlinear function. The unscented transformation tells us that to estimate the mean and variance of $f(x)$ we do not need to have a full scale Monte Carlo simulation. Instead we can choose a small set of carefully designed sigma points around μ, then apply the function f on those sigma points and take their weighted mean and variance to approximate the mean and variance of $f(X)$. The rule to select the sigma points $\mathcal{X}$ are

$$\begin{aligned}
\mathcal{X}_0 &= \mu, \\
\mathcal{X}_i &= \mu + \left(\sqrt{(L+\lambda)\Sigma}\right)_i, \quad i = 1, \ldots, L, \\
\mathcal{X}_i &= \mu - \left(\sqrt{(L+\lambda)\Sigma}\right)_{L-i}, \quad i = L+1, \ldots, 2L,
\end{aligned}$$

where $\left(\sqrt{(L+\lambda)\Sigma}\right)_i$ is the ith column of the Cholesky decomposition of $(L+\lambda)\Sigma$. Then we propagate the selected points by the nonlinear function such that $\mathcal{X}_i^f = f(\mathcal{X}_i)$. In order

to find the weighted mean and variance

$$\mu^f = \sum_{i=0}^{2L} \omega_i^m \mathcal{X}_i^f,$$
$$\Sigma^f = \sum_{i=0}^{2L} \omega_i^v (\mathcal{X}_i^f - \mu^f)(\mathcal{X}_i^f - \mu^f)^T,$$

the weights are also carefully designed as $\omega_0^m = \frac{\lambda}{\lambda+L}$, $\omega_0^v = \omega_0^m + (1 - \alpha^2 + \beta)$ and $\omega_i^m = \omega_i^v = \frac{1}{2(\lambda+L)}$, $i = 1, \ldots, 2L$.

We need to feed in three tuning parameters α, β and κ. A normal set of values are $\alpha = 10^{-3}$, $\kappa = 0$ and $\beta = 2$. Based on those, $\lambda = \alpha^2(L + \kappa) - L$.

A couple of points needed to be mentioned.

1. The size of the sigma points is $2L + 1$, where L is the exact dimension of the random vector X. Hence the number of Monte Carlo points is VERY SMALL.

2. Combination of α and κ can control the spread of the sigma points.

Now back to our model (4.2) in Chapter 4, we can describe the Unscented Kalman filter upon the unscented transformation.

1. Given the filtering mean and variance of $\boldsymbol{Z}_t | \boldsymbol{X}_{0:t}, \boldsymbol{\theta}$ are μ_t and Σ_t, and our objective is to find the mean and variance of $\boldsymbol{Z}_{t+1} | \boldsymbol{X}_{0:t+1}, \boldsymbol{\theta}$.

2. Define an augmented random vector $\boldsymbol{Z}_t^* = (\boldsymbol{Z}_t, \boldsymbol{\epsilon}_t)$, which has a dimension of $2p + q$, where p and q are defined with the model (4.2). This augmented random vector $\boldsymbol{Z}_t^*$ has the mean and variance of $\begin{pmatrix} \mu_t \\ \mathbf{0} \end{pmatrix}$ and $\begin{pmatrix} \Sigma_t & \mathbf{0} \\ \mathbf{0} & I_{p+q} \end{pmatrix}$. In practice, we find that to draw the sigma points based on variance I_{q+2p} and treat Σ_t as one part of the nonlinear transformation is better than to directly draw sigma points based on Σ_t. Indeed, we use this approach in our practice.

3. Apparently, $\boldsymbol{\Lambda}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t) + \boldsymbol{\Omega}_{\boldsymbol{\theta}}^{1/2}(\boldsymbol{Y}_t)\boldsymbol{\epsilon}_t$ is a vector of nonlinear functions on the augmented random vector $\boldsymbol{Z}_t^*$, and this function projects a $2m + n$-dimensional vector into a $m + n$-dimensional vector. Hence, we can find the approximated mean and variance of $\boldsymbol{Y}_{t+1}|\boldsymbol{X}_{0:t}, \boldsymbol{Z}_t$ by the unscented transformation as we described above. The obtained mean and variance are the updated filtering mean and variance of $\boldsymbol{Z}_{t+1}|\boldsymbol{X}_{0:t+1}, \boldsymbol{\theta}$

The above approach combines the steps of *prediction* and *updating* in a conventional Kalman-style filtering framework. It is not obvious but the correlations between the observed states and latent states are actually included in the nonlinear transformation $\boldsymbol{\Lambda}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t) + \boldsymbol{\Omega}_{\boldsymbol{\theta}}^{1/2}(\boldsymbol{Y}_t)\boldsymbol{\epsilon}_t$.

Appendix D

Positive Definiteness of the Correlation Matrix

Given the hierarchical design to obtain the correlation matrix in Section 3.2, we can write the Brownian motions of $(\boldsymbol{B}_t^S, \boldsymbol{B}_t^V, B_{0t}^V)$ as

$$\begin{pmatrix} \boldsymbol{B}_t^S \\ \boldsymbol{B}_t^V \\ B_{0t}^V \end{pmatrix} = \boldsymbol{L} \begin{pmatrix} \boldsymbol{B}_t^\eta \\ B_{0t}^Z \\ \boldsymbol{B}_t^\varepsilon \\ B_{0t}^V \end{pmatrix}$$

where $\boldsymbol{B}_t^\eta = (B_{1t}^\eta, \ldots, B_{qt}^\eta)$, $\boldsymbol{B}_t^\varepsilon = (B_{1t}^\varepsilon, \ldots, B_{qt}^\varepsilon)$ and

$$\boldsymbol{L} = \begin{pmatrix} \tilde{\boldsymbol{\rho}}\tilde{\boldsymbol{\omega}} & \tilde{\boldsymbol{\rho}}\boldsymbol{\omega} & \boldsymbol{\rho}\tilde{\boldsymbol{\tau}} & \boldsymbol{\rho}\boldsymbol{\tau} \\ \mathbf{0} & \mathbf{0} & \tilde{\boldsymbol{\tau}} & \boldsymbol{\tau} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & 1 \end{pmatrix}$$

where

$$\begin{aligned}
\tilde{\boldsymbol{\rho}}\tilde{\boldsymbol{\omega}} &= \text{diag}(\sqrt{1-\rho_1^2}\sqrt{1-\omega_1^2}, \ldots, \sqrt{1-\rho_q^2}\sqrt{1-\omega_q^2}) \\
\tilde{\boldsymbol{\rho}}\boldsymbol{\omega} &= (\sqrt{1-\rho_1^2}\,\omega_1, \ldots, \sqrt{1-\rho_q^2}\,\omega_q)' \\
\boldsymbol{\rho}\tilde{\boldsymbol{\tau}} &= \text{diag}(\rho_1\sqrt{1-\tau_1^2}, \ldots, \rho_q\sqrt{1-\tau_q^2}) \\
\boldsymbol{\rho}\boldsymbol{\tau} &= (\rho_1\tau_1, \ldots, \rho_q\tau_q)' \\
\tilde{\boldsymbol{\tau}} &= \text{diag}(\sqrt{1-\tau_1^2}, \ldots, \sqrt{1-\tau_q^2}) \\
\boldsymbol{\tau} &= (\tau_1, \ldots, \tau_q)'
\end{aligned}$$

Since $(\boldsymbol{B}_t^{\eta}, B_{0t}^{Z}, \boldsymbol{B}_t^{\varepsilon}, B_{0t}^{V})$ are independent Brownian motions. The variance-covariance matrix of $(\boldsymbol{B}_t^{S}, \boldsymbol{B}_t^{V}, B_{0t}^{V})$ is

$$\boldsymbol{L}\boldsymbol{L}'t.$$

Hence for $\forall \boldsymbol{a} \in \mathbb{R}^{2q+1}$ and $\boldsymbol{a} \neq \boldsymbol{0}$, we can easily find $\boldsymbol{a}'\boldsymbol{L}\boldsymbol{L}'\boldsymbol{a} > 0$. Note $\boldsymbol{a}'\boldsymbol{L}$ is a $2q+2$ vector, where the first q elements should have at least one larger than zero, since $\tilde{\boldsymbol{\rho}}\tilde{\boldsymbol{\omega}}$ is a non-zero matrix.